



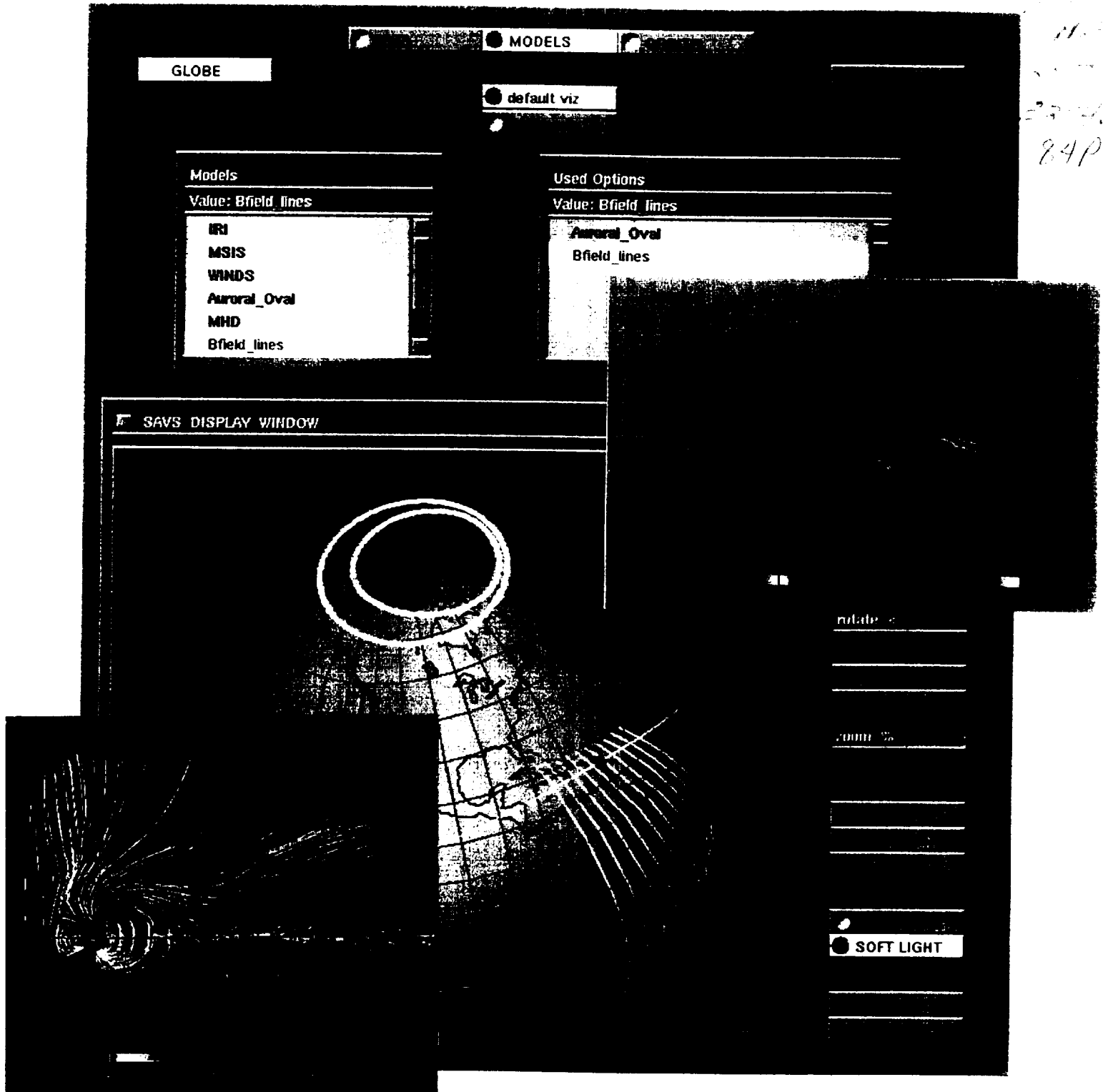
SAVS

A Space and Atmospheric Visualization Science System

CR-189390

Contract No. NAS5-32337

Final Report - October 31, 1994



Prepared by

E. P. Szuszczewicz, P. Blanchard, and A. Mankofsky

With contributions from

C. Goodrich, D. Kamins, R. Kulkarni, D. McNabb, and M. Moroh

SAIC

Preface and Acknowledgments

This report documents the concepts, development efforts, and the final system architecture of an integrated data acquisition, analysis and visualization system called SAVS. The effort was supported under a three-year contract NAS5-32337 within NASA's Applied Information Systems Research Program and its fundamental objective focused on the need to enhance scientific productivity within the Space and Atmospheric Science communities. Within the effort, considerable attention was placed on the overall architecture, with the intent being ease, functionality and extensibility. System portability received substantial attention as well as the need for easy access to local and remote data sets and models. CDF, HDF and netCDF readers were designed, built and implemented, a library of general utility space science models was constructed, special modules were developed for NASA application needs, data browse and access tools were implemented, and interactive interpreter capabilities were included. And finally, "typical" NASA problems were addressed as demonstrations of the overall system capability to handle different types of models, data formats, and scientific needs.

The overall program was designed and directed by SAIC's Laboratory for Atmospheric and Space Science with the help of Advanced Visual Systems Inc., and subcontractual support of the University of Maryland. Along the three year trail of development and test activities there were many important contributions from individuals not directly associated with the tasks but interested in its goals. We acknowledge the inspiring leadership of J. Bredekamp in the formulation and management of the AISRP activities and in his support of SAVS. We are grateful to G. Mucklow for his handling of the program workshops and support for a special session on visualization techniques at the Spring 1993 meeting of the American Geophysical Union. Among the many technical contributions we are pleased to extend our thanks to J. Green, J. King, B. Jacobs, R. McGuire, and R. Candey at NASA/NSSDC, to P. Esdale and L. Gelberg at AVS, Inc., and to D. Corley and J. Coyne at SAIC. Without all their individual contributions SAVS would not have come to its current stage of development and overall utility.

TABLE OF CONTENTS

Title Page	i
Preface and Acknowledgements	ii
Table of Contents	iii
1. INTRODUCTION	1
1.1 Overview of Objectives and Contents.....	1
1.2 The Problems in Space and Atmospheric Sciences.....	2
1.3 Visualization: The Pathway to Enhanced Scientific Productivity	3
1.4 Illustrations of Visualization Techniques.....	4
1.5 The Need for an Integrated System Like SAVS	11
2. THE SAVS ARCHITECTURE AND ITS SUBSYSTEMS.....	13
2.1 An Integrated System	13
2.2 Uniqueness of the Architecture and Its End-to-End Approach	13
2.3 The Visualization Software	14
2.4 The <i>Push-Button</i> Front End Architecture.....	16
2.5 Data Access.....	17
2.6 Visualization Applications to Mission Scenarios.....	19
2.7 Summary Comments on the Overall System	21
3. GENERAL INTERFACE FUNCTIONALITIES.....	23
3.1 Concourses	23
3.1.1 The <i>DATA</i> Concourse Mode.....	26
3.1.2 The <i>MODELS</i> Concourse Mode	27
3.1.3 The <i>ORBIT</i> Concourse Mode	28
3.2 Display Window	29
3.3 Pull Down Menu (i.e., Menu Bar).....	30
4. HANDS-ON APPLICATIONS	35
4.1 Data-Model Comparison: Compare Sai Auroral Image & Feldstein Auroral Model.....	35
4.2 Integrating Models & Math Module with Stack Plots: Sigma Calculations of Field-Line Integrated Conductivities.....	37
REFERENCES	41
APPENDICES (SAVS MODULE DESCRIPTIONS)	
A. DATA	43
B. MODELS.....	53
C. FILTERS	64

1. INTRODUCTION

1.1 Overview of Objectives and Contents

The research environment faced by space and atmospheric scientists in the coming years is characterized by unprecedented volumes of new data, by ever-increasing repositories of unexploited mission files, and by the widespread use of empirical and large-scale computational models needed for the synthesis of understanding across data sets and discipline boundaries. The effective analysis and interpretation of such massive amounts of information demands a new approach with visualization at the heart of the requirements for a much-needed enhancement in scientific productivity as we face these new challenges. This report opens with a perspective on the problem as it crosses discipline boundaries from solar physics to atmospheric and ocean sciences, with an attempt to introduce visualization as a new approach to scientific discovery and a tool which expedites and improves our insight into physically complex problems. A set of simple illustrations demonstrates a number of visualization techniques and the discussion emphasizes the trial-and-error and search-and-discovery modes that are necessary for the techniques to reach their full potential. Further discussions point to the importance of integrating data access, management, mathematical operations, and visualization into a single system. This is the integrated approach of SAVS.

SAVS (a Space and Atmospheric Visualization Science System) is a complete system with user-friendly functionality that employs a *push-button* software environment that mimics the logical scientific processes in data acquisition, reduction, analysis, and visualization. All of this is accomplished without requiring the user to have a detailed understanding of the methods, networks, and modules that link the tools and effectively execute the functions.

SAVS provides:

- 1) a customizable framework for accessing a powerful set of visualization tools based on the popular AVS visualization software with hooks to PV-Wave,
- 2) a set of mathematical and statistical tools,
- 3) an extensible library of discipline-specific functions and models (e.g., MSIS, IRI, Feldstein Oval, the IGRF and Tsyganenko B-field models, etc.), and
- 4) capabilities for local and remote database access.

The system treats scalar, vector, and image data, and runs on most common Unix workstations.

After a general discussion of the new and evolving problems in space and atmospheric sciences and important perspectives to be gained through visualization, we present a description of SAVS and its components. That description is followed by several applications based on generic research interests in interplanetary and magnetospheric physics (IMP/ISTP), active experiments in space (CRRES), and mission planning focused on the Earth's thermospheric, ionospheric and mesospheric domains (TIMED). The final chapters provide a user-oriented description of interface functionalities, hands-on operations, and customized modules, with details of the primary modules presented

in the Appendices. The overall intent of the report is to reflect the accomplishments of the three-year development effort and to introduce potential users to the power and utility of the integrated data acquisition, analysis and visualization system called SAVS.

1.2 The Problems in Space and Atmospheric Sciences

Space and atmospheric science is shifting in emphasis from single exploratory investigations to concentration on collaborative missions of large-scale system phenomena producing unprecedented volumes of data with unmatched temporal, spectral and spatial resolution. The multi-parameter, multi-platform databases will challenge our intellectual capabilities to absorb, synthesize, and effectively analyze the measurements, and impose new demands on software and hardware systems as we know them today.

The large-scale numerical codes intended to model, interactively analyze, and ultimately predict the observed phenomena are of an equal challenge, requiring their implementation on massively parallel computers with performance characteristics approaching the teraflop range. These codes, whether they address individual disciplines like solar, heliospheric, or atmospheric physics, or the coupling of solar-terrestrial processes that control our weather, have commonality in their complexities. They all involve three-dimensional time-dependent geometries, large variations in temporal and spatial scales, complex physical interactions, and most often non-linear dynamics.

If we are to meet the challenge of the National Academy of Sciences to be able to predict the Earth's environment in the context of its position in the solar system and in the context of global change, we must: 1) develop the systems which can efficiently gather and process the data, 2) access, compare and *tune* the appropriate models, and 3) integrate the results into a predictive capability. Here the integration of human intelligence with hardware and software systems and appropriately designed visualization tools plays a pivotal role if these ambitious goals are to be achieved and a predictive capability developed.

In the domain of solar and heliospheric physics we expect 3D magnetohydrodynamic (MHD) models to be among the basic tools for a coherent hierarchical approach to the study of solar processes and their control of the global heliosphere from its coronal origins to its interactions with the Earth's bow shock and the interstellar medium. The origins of the solar wind as well as coronal mass ejections (CMEs) must be modeled to help interpret ground-based and spacecraft data such as that from the Solar Heliospheric Observatory (SOHO). Similarly, the interaction of solar wind streams, the propagation of CMEs, and the structure of the heliosphere out of the ecliptic must be modeled and compared with data from the Ulysses spacecraft. And the distant interaction of the solar wind with the local interstellar medium, leading to the formation of the solar wind termination shock, the heliopause, and the heliospheric bow shock, must be modeled to help interpret data from the Pioneer and Voyager spacecraft now reaching towards the edge of the solar system.

The work of the solar and heliospheric communities includes the requirements of magnetospheric physicists who will be more intent in looking out to the interplanetary medium and dynamic solar phenomenologies to understand the cause-effect relationships in their discipline area of geospace. Upper atmospheric and thermospheric

physicists will continue to be more keenly aware of the need for an understanding of ionospheric coupling mechanisms and the controls from the magnetosphere and the heliosphere above, and the stratosphere below. And scientists studying the Earth's ionospheric-thermospheric-mesospheric system will be looking to Mars, Venus and beyond for comparative studies of large-scale planetary environments that not only form an intrinsic element in solar-system exploration but provide unique opportunities to determine the limits of Earth-based concepts and extend our fundamental understanding of the processes under study.

Atmospheric and ocean scientists will be looking for improved global-scale measurements of sea state and surface winds to better understand air-sea interactions, heat transfer mechanisms, weather, and climate. And the hydrologic cycle will be an experimental and theoretical focus because of its important role in creating and modulating the various climatic zones on the Earth. The global circulation models will therefore require the integration of large volumes of environmental data, including rain-rate and rain-distribution data from ground-based radars, rain gauges, and spaceborne sensors like those included in the Tropical Rainfall Measurement Mission (TRMM).

The increased focus on large-scale system phenomena, the cross-disciplinary nature of many investigations, and projections of increased volumes of *in situ* and remote sensing data are all elements in the future of Earth and Space Sciences. These projects range from new astronomical observatories such as the Space Telescope, planetary orbiters such as Galileo and Cassini, the multi-spacecraft International Solar Terrestrial Physics (ISTP) and TIMED missions, and the EOS mission to planet Earth. ISTP alone is expected to accumulate almost a gigabit of data per second.



To be effective in their work scientists require a data analysis environment with sophisticated interactive graphics tools based on advanced visualization techniques. Such techniques represent one of the most effective ways to *compress* complex data into a visually organized form which is optimized for analysis and interpretation. The scientists also require easy-to-use mathematical and image processing tools. In addition, they will need tools to assist in obtaining data from remote archives as well as to access the results of empirical and numerical models to correlate with the data and assist in analysis and interpretation. To be most effective, these tools must be integrated into a single user-friendly system. This is the SAVS system.

1.3 Visualization: The Pathway to Enhanced Scientific Productivity

Visualization is at the heart of our requirements for the much needed enhancement in scientific productivity as we face the grand challenge of large-scale system phenomena in space and Earth sciences. This is not to underplay the important roles of remote data access or easy-to-use mathematical tools, because they are the fundamental elements with which the visualization process begins.

In itself scientific visualization is a rapidly evolving field. But the reality is that we have been applying visualization techniques in one form or another during our entire scientific careers. What has changed has been the magnitude of the problems we are addressing, the sensitivity and dimensionality of our instruments, the volume of our database, and the hardware, software, and computational tools available to interpret and assimilate the results. Visualization is not new. We are simply in a new era that requires a dramatic change in our approach and our attitudes toward data handling and

analysis. This attitude must recognize that visualization is not a pretty picture but rather a process of discovery that links the knowledge of the discipline with the knowledge of visualization tools. In the past what may have been called visualization was really *data display*, that is, the graphical rendering of the data when we knew what it meant or when we knew what we wanted it to say. On the other hand, the *visualization process* as we have come to know it today is the intersection of intellectual pursuit with a rendering of the data or model results which leads to discovery. The rendering involves the use of color, intensity, transparency, texture, animation and many other techniques which can convey a tremendous amount of information in a single image and in a short period of time. The overlay of images with varying degrees of transparency and texture, or the superposition of scalar and vector fields, or a sequence of images (*i.e.*, animation) can yield more information than the sum of the parts. This allows our intellectual capabilities to parallel process the results as no machine can do, and couples the knowledge of our discipline with the knowledge of the visualization tools.

Emphasizing its functional aspects, visualization can be thought of as having two major components: exploration and presentation [Keller and Keller, 1993]. In the exploration mode we search the data or model results for new relationships or insights into complex phenomena. Often this means trial-and-error representations of the data and/or model results, requiring interactive adjustments of the data or the image and the direct participation of the scientist's knowledge of the discipline. It is in this mode that we emphasize visualization as a *process* rather than the graphical end-product of the "presentation" mode in which the arrangement and display of the data is for publication and the benefit of others.

It must be emphasized that the exploitation of visualization technologies and the subsequent enhancement in scientific productivity require a somewhat non-traditional approach to the handling and rendering of data and model results. Unfortunately, many scientists have been conditioned to regard data and model outputs as entities with inviolate properties that ultimately dictate the use of standard displays and presentation formats for the data. Such rigid thinking precludes the exploitation of visualization techniques and undermines the exploration phase where trial and error in the coupling of knowledge of the discipline with the knowledge of the visualization tools is the pathway to discovery.

1.4 Illustrations of Visualization Techniques

Recognizing that visualization is a process involving tools that include the use of color, transparency, texture, etc., along with rotation, pan, zoom, and animation [see *e.g.* Friedhoff and Benzon, 1989; and Nielson et al., 1990], this section presents simple illustrations of several visualization techniques to highlight their utility in unfolding complex data, in developing quick insights into a problem, and in conveying large amounts of information in short periods of time.

Color vs isolines and multi-parameter overlays. Color is one of the simplest tools for visualizing data. It finds extended use in identifying morphologies in a scalar field. Isolines can do this as well, but color can identify morphologies when isolines fail. This is illustrated most simply with the blue-to-red color rendering of the scalar field of numbers ranging from 0 (blue) to 9 (red) in Figure 1. This might be a field of temperature measurements over a land mass or over the ocean. It might also represent plasma density distributions under disturbed conditions in the polar cap ionosphere. In

any case, the visualization task must unfold unique morphological features (*i.e.*, concentrations, rarefactions, etc.). Isolines would fail in this application, since they would yield a tangled net of spaghetti that provides no simple organization of the data. On the other hand, the use of color immediately identifies a hot spot of 9s near the center of the image in of Figure 1.

Color has found great utility in a number of disciplines where there is the need for multi-parameter overlays in endeavors focused on understanding a phenomenon rather than a single data set. A typical application is in thermospheric physics, where two dimensional displays in latitude and longitude can, for example, depict atomic oxygen densities in color, molecular nitrogen densities as isolines, and winds with vectors (see Figure 2). Such an image is effective, but only at a given altitude. Understanding the full three-dimensional, time-dependent aspect of thermospheric circulation and its coupling to the global-scale ionosphere under magnetic storm conditions requires more advanced visualization tools than those demonstrated in Figure 2. Such tools include isosurfaces, streamlines, transparency, texturing, and animation, all of which are available in a number of commercial software packages (*e.g.*, AVS from AVS, Inc., Data Explorer from IBM, etc.), and some of which have been implemented already in developmental applications [*e.g.*, Foster et al., 1994; Twiddy et al., 1994; and Gekelman, 1994].



Figure 1. A color rendering of a sample two-dimensional array of numbers (0 [blue] through 9 [red]) mimicking any scalar field of measurements where morphologies or topologies need to be determined. The color representative of the numerical field clearly reveals a concentration of high-scale measurements (9's) near the center of the image, and a random distribution elsewhere .

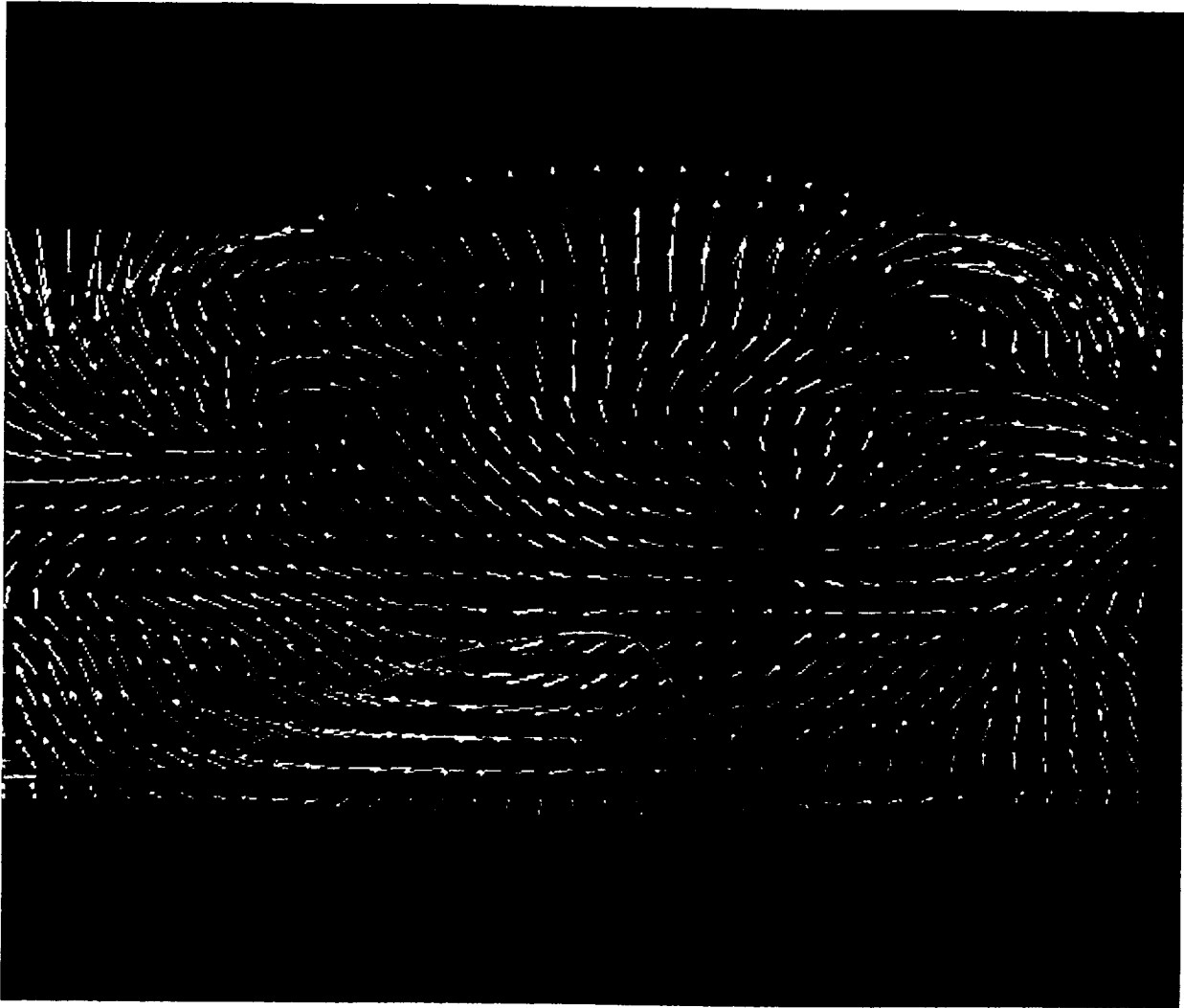


Figure 2. A SAVS illustration of color overlays which superimpose geophysical parameter sets in a study of the Earth's thermosphere. Image shows the global distributions of winds (white vectors), molecular nitrogen densities (color-coded isolines), and atomic oxygen densities (full color-coded cut-plane) at 300 km as specified by the MSIS model [Hedin, 1991].

Isosurfaces and the need for a non-traditional approach to data presentation. The exploratory nature of the visualization process requires that scientists free themselves from rigid historical views on the display and presentation of their data. When this is done the opportunity presents itself for entirely new perspectives on the data and associated insights into the prevailing physical principles. This is illustrated below with a non-traditional approach to the representation of the Earth's geomagnetic field using the International Geomagnetic Reference Field model [IGRF, available from the National Geophys. Data Ctr., Boulder, CO].

Typically there are two standard representations of the Earth's field. The first involves two-dimensional isocontours (*i.e.*, isolines) of B-field values at a fixed altitude. The second is a fieldline representation that demonstrates the field's dipolar nature. A non-traditional approach is to view the geomagnetic field with a three-dimensional isosurface of sequential B-field values. One such image for the B=0.5 and 0.265 Gauss isosurfaces

is presented in Figure 3, which shows the $B=0.5$ surface (yellow) confined to the polar regions. The $B=0.265$ surface is more global, with the image showing it furthest from the Earth near the north and south polar regions, closest to the Earth at mid- and equatorial latitudes, and intersecting the Earth in the South American/South Atlantic region. This intersection locates the well-known South Atlantic Anomaly where the field at the surface of the Earth has anomalously small values and where, as a consequence, energetic particles are known to penetrate to very low altitudes [Ratcliffe, 1972]. This picture presents an intuitive insight into the causality, recognizing that the isosurface represents an approximation to the magnetic pressure ($B^2/8\pi$) and that energetic particles have higher probabilities of atmospheric entry in the south Atlantic pressure well that exists naturally in the Earth's B-field topology. One can only speculate on the impact that such a visualization tool may have had in early studies of the Earth's field and its control of charged particle dynamics.

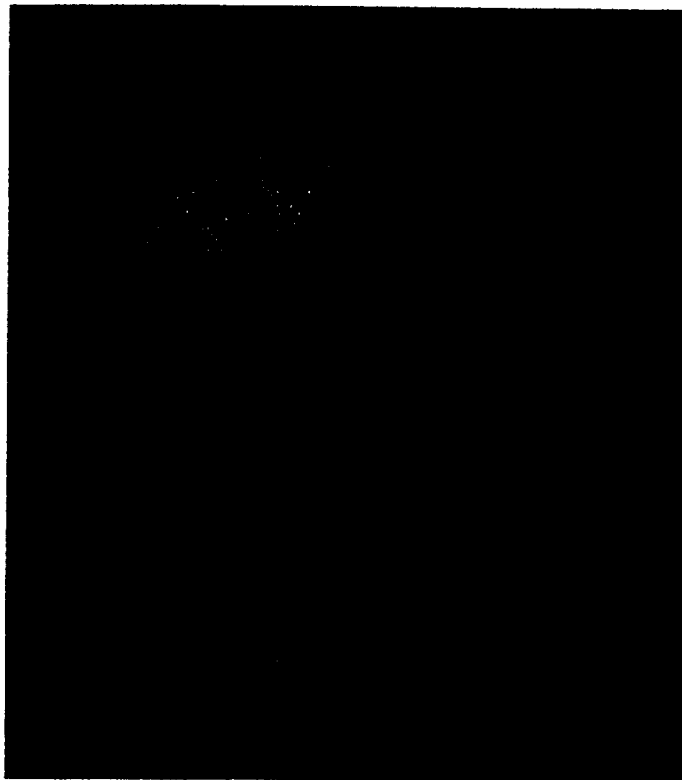


Figure 3. SAVS illustration of a non-traditional approach to rendering data and modeling results. Visualizing the Earth's geomagnetic field with an isosurface application to the IGRF model immediately reveals the phenomenological domain called the South Atlantic Anomaly. See text for discussion.

The visualization process in data reduction and model-measurement comparisons. The products of visualization tools are not always representations of data and model runs. They often provide tests of model or algorithm integrity and time-saving views of an experiment scenario that facilitate determination of the shortest path in the process of data reduction and analysis. These aspects of visualization are illustrated in an application to a rocket-borne chemical release experiment in the NASA/CRRES program that involved *in situ* and ground-based systems and aspects of coupled phenomena along geomagnetic field lines.

The first panel of Figure 4 shows the rocket trajectory along with an array of B-field vectors determined by the IGRF. The obvious discontinuity in the B-field representation

revealed a problem in an interpolation module that may have gone undetected without this simple visual aid.

The second panel of Figure 4 illustrates the application of visualization tools to help understand the actual configuration of the experiment. In the simplest description of the experiment scenario, a Ba/Li mixture in a sealed canister was separated from the rocket's diagnostics payload and the vaporized gases were released on the upleg portion of the trajectory at an altitude near 180 km. The colored disk in the panel is a color-coded model depiction of the cloud's ion density in the plane perpendicular to the cloud's bulk velocity at a very early phase of the expansion process. The red line is the suborbital trajectory, and the long and short vectors, respectively, are the B-field and payload velocity at the point of the release. The cone-shaped object in the figure is the projection of a ground-based hf heater beam intended to intercept the chemical cloud at its ionospheric altitude, excite the electrons, and induce enhanced optical emissions and plasma instability processes.

The initial process of understanding the actual configuration of the experiment involved the following questions:

- 1) What were the magnitude and directions of the cloud's bulk velocity relative to the geomagnetic field?
- 2) How well was the chemical release coupled along the B-field to the diagnostic payload?
- 3) How effectively did the heater beam intercept the cloud?
- 4) Was it possible to observe expanding cloud ions on the downleg portion of the trajectory?

The first three questions were answered using rotation and zoom tools, with some of the image products presented in the middle and right-hand panels of Figure 4. The last question was answered with yet another rotation that projected an image perpendicular to the plane of the trajectory, with the downleg portion of the trajectory in the foreground. That image, presented in the right-hand panel of Figure 4 shows that the B-field at the release site did not intersect the downleg trajectory. A simple hand-scaling of altitude and range showed that the flux tube passing through the release missed the downleg trajectory by 13 km, a sufficiently great distance to minimize interests in searching the data for possible late-time ion signatures of the expansion process on the downleg trajectory.

The entire exercise of answering the questions listed above required less than five minutes. Without the visualization tools, the answers would have required the development of a number of mathematical algorithms and a sequence of stack plots. That exercise could have taken 3 to 5 days, depending on the proficiency of the programmer.

This illustration is one of many that could have been presented to highlight visualization as a time-saving and insightful element in the process of data reduction and analysis. This is especially true in situations that require co-registration of spaceborne *in situ* and

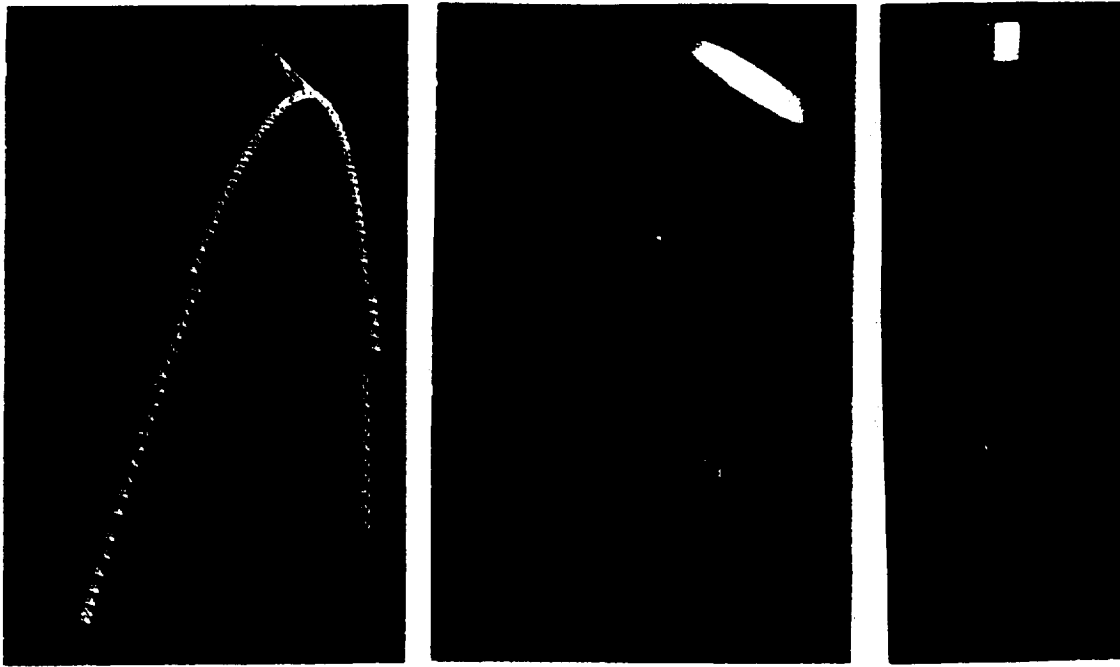


Figure 4. SAVS illustration of visualization as an integral process of understanding an experiment configuration and optimizing the approach to data reduction and analysis. See text for details.

remote sensing techniques supported by ground-based diagnostics. For example, the NASA TIMED mission will rely heavily on spaceborne and ground-based remote sensing techniques in its attempt to understand the coupling of the ionospheric, thermospheric and mesospheric domains and the influence of magnetospheric processes. The three-dimensional mapping of gravity wave influences, dynamic changes in neutral composition, field-line integrated conductivities, and penetrating electric field events are among the many issues that will need to be addressed. This will require considerable care in space and time correlations along with co-registration of instrument fields-of-view with coupled regions of the mesosphere, the lower thermosphere, and the D, E, F₁, and F₂ regions of the ionosphere.

On the rendering of three-dimensional processes with isosurfaces, cut-planes and transparencies. The problem of co-registration and space-time correlations among measurement platforms and instrument types can be particularly acute in investigations of the solar wind-magnetosphere-ionosphere (SW-M-I) system. Such investigations involve an incredible volume of space with prevailing phenomena that span the spatial spectrum from meters to several Earth radii. The coupling mechanisms within the SW-M-I system are at the center of the NASA ISTP and the NSF GEM programs, with a focus on the flow of energy and momentum into and through the system. The difficulties include the complexities of the interacting phenomena, the large set of controlling parameters, and the coupled domains of varying dimensions.

The utility of several visualization tools in the study of the SW-M-I is illustrated here through applications to the MHD model of J. Lyon [Fedder and Lyon, 1987] in Figure 5. This figure has been created to address a Gedanken experiment that attempts to determine the flow of momentum into specified regions of space under steady-state and dynamic environments. The upper panel presents the following elements: 1) the inner

boundary of the computational domain at $3.5 R_E$ in red, 2) open and closed field lines in white, and 3) a pair of transparent isosurfaces (*i.e.*, isobars of constant plasma pressure) with a color scale that represents the magnetic pressure term $B^2/8\pi$ everywhere on the isobaric surface. (Transparency is used to show the B-field structure within the isobaric surface).

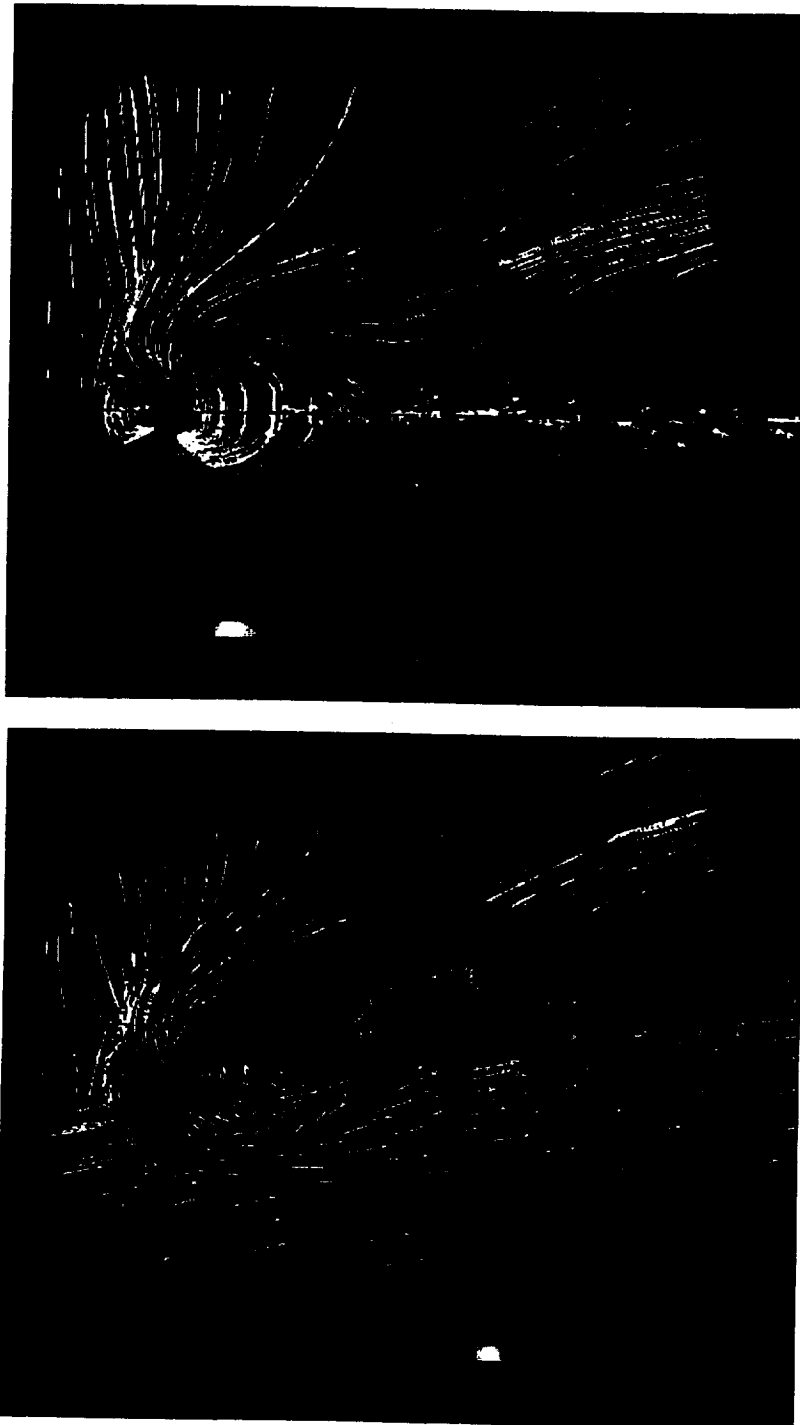


Figure 5. SAVS illustration of imaging and visualization perspectives in addressing magnetospheric problems. Based on the MHD model of Fedder and Lyon [1987], this figure includes representations of the computational grid along with plasma and magnetic field pressure, bulk velocities, plasma densities, and open/closed field lines. See text for complete discussion.

The lower panel is a two-angle rotation (*i.e.*, pitch and roll) of the top image with the transparency of the isobaric surfaces eliminated. The cut-plane, which is perpendicular to the plane of the B-field lines, presents: 1) bulk plasma velocity vectors in white, 2) the plasma density in a slightly transparent color code, and 3) the spider web computational grid of the MHD model which provides varying degrees of resolution in the computational domain. This image sets the stage for an improved understanding of the topologies and morphologies in the overall system and for the calculation of momentum and energy flows into and out of critical domains. In this case the Gedanken experiment is concentrated on the isobaric surfaces whose topology we see is controlled by the lowest-latitude open field lines in the nightside hemisphere.

These are just two of many possible images that can be at the researcher's disposal. They clearly display the morphological relationships among the computational grid, the field configuration, the kinetic and magnetic pressures, the plasma density, and the bulk flow velocities — all elements critical to the calculation of energy and mass transfer. Without such images, insight and understanding of the problem would be impaired.

1.5 The Need for an Integrated System Like SAVS

Recognizing that visualization is a process that enhances insight into the reality of physically complex problems is just the beginning for enhanced scientific productivity in the expanding missions of space and Earth sciences. Visualization is indeed a condition without which we cannot effectively address the grand challenges of the coming decade. Scientific productivity demands an interactive data analysis environment built around advanced visualization tools, because such an approach represents one of the most effective ways to compress complex data into a visually-organized form that is optimized for analysis and interpretation.

While it is generally agreed that easy-to-use visualization and data handling tools are vital to efficient and insightful progress in science, the applications development environment has been characterized as fragmented, with little overall direction or coordination [Botts, 1992]. It has been argued that the primary bottleneck is the lack of adequate software which allows the scientist to take advantage of today's computing power and interactively visualize and analyze his/her data within the complex computing environment. The end result is that the largest fraction of the scientific community is not using the visualization capabilities available today because (according to Botts): 1) the tools are not extensible or are too inflexible, 2) it is too difficult to input data, 3) the tools do not adequately link visualization and analysis, and 4) the tools do not meet scientific expectations. It is also agreed throughout the scientific community and effectively argued by Botts that the main general areas needing consideration include:

- 1) Integration of visualization with data management and analysis,
- 2) A framework of flexibility without complexity, and
- 3) The development of an architecture focused on the scientist as the end user.

This requires that the tools meet the actual needs of the scientist, be simple and intuitive to use, and be logical to the scientist rather than to a computer specialist. These were the SAVS design criteria.

For SAVS or any similar system to reach its full potential there is the intrinsic need for scientists and engineers to open their minds to the broad spectrum of capabilities imbedded in the visualization tools that are made available. As discussed earlier, many scientists have been conditioned to regard data and model outputs as entities with inviolate properties that dictate the use of standard displays. This attitude must be abandoned for the full potential of visualization to be realized. It is a rapidly developing field, that holds great promise. But like any resource, it must be effectively mined.

2. THE SAVS ARCHITECTURE AND ITS SUBSYSTEMS

2.1 An Integrated System

SAVS represents an integrated system concept with a push-button approach to the access and implementation of tools necessary to:

- search for, acquire, and read data,
- establish satellite tracks, their relevant parameters, and their co-registrations with allied multi-spacecraft and ground-based diagnostics,
- interactively interpret and analyze the findings,
- compare the results with mature and/or developing models, and
- creatively visualize the overall process and resulting scientific products.

Much of the SAVS effort has been devoted to the development and implementation of a user-friendly architecture focused on ease and functionality. Within our push-button approach, we have embedded a logical interactive data handling and analysis methodology, a mathematical toolbox, an interactive interpreter, and the AVS visualization software [Upson et al., 1989]. As part of our effort, we have customized AVS, its tools, and its operation for the NASA scientific user environment. Within this framework we have created custom functions, and integrated local and remote database access, data translation, and formatting tools in order to provide scientific data manipulation, analysis, and display capabilities not available in the basic AVS system.

2.2 Uniqueness of the Architecture and Its End-to-End Approach

The SAVS architecture recognizes that the scientist and mission design engineer work routinely with data, theoretical models, and orbits (including orbital parameters, payload configuration, instrument fields-of-view, coordinated ground stations, *etc.*). At any given time they may work exclusively with models, data, or orbits, or combinations of all three. The most likely scenario involves an interactive combination of the three, filtered through a mathematical interpreter and displayed through a broad set of visualization tools. This end-to-end approach of SAVS is illustrated in Figure 6, where the products of the SAVS system serve a range of NASA needs including: 1) data synthesis; 2) mission planning; 3) science and engineering trade studies; 4) data-model comparisons; and 5) model test and validation.

In mimicking the thought processes of the NASA scientist, the SAVS architecture has three entry concourses for daily activities: DATA, MODELS, and ORBITS. By a push of a button in the SAVS front end scientists may enter any one of the three concourses and begin a procedure involving SAVS-guided functionality whereby they can exercise an unlimited number of loops in any given concourse. With this approach the scientists can treat and overlay multiple model runs or datasets, and visualize the results in any number of formats. This SAVS concourse methodology and its generic products are illustrated in Figure 6.

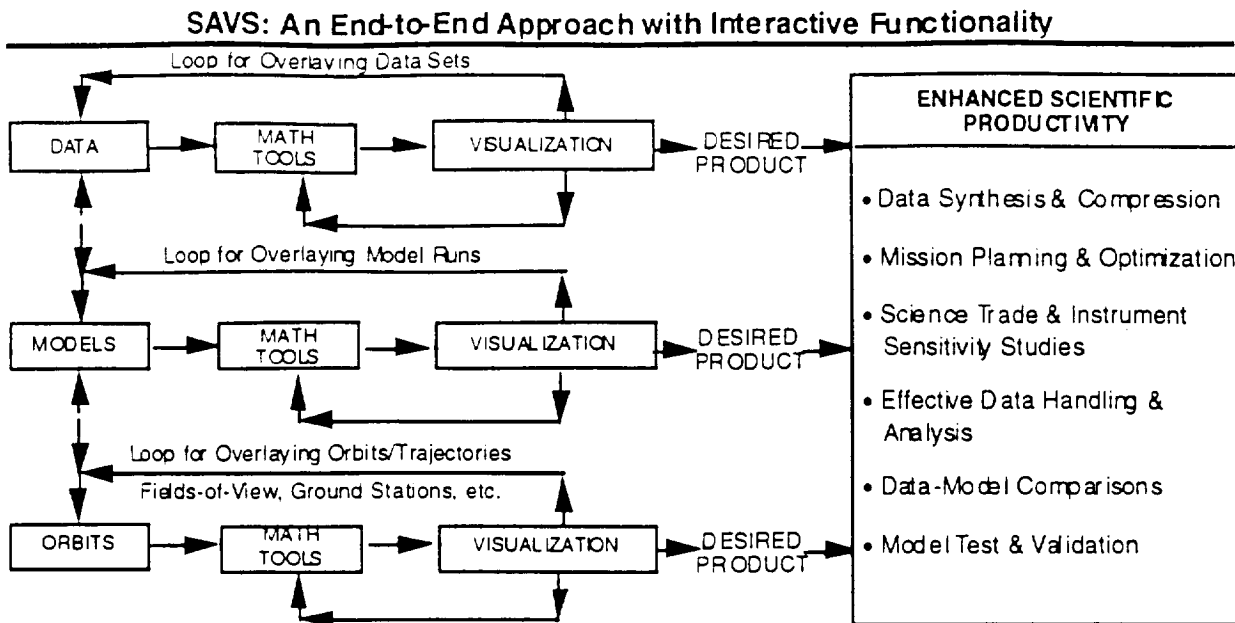


Figure 6. The SAVS entry concourses of **DATA**, **MODELS** and **ORBITS** allow focus while providing flexibility for user-defined overlays of data, model and experiment parameters in any number of visualization formats. The interactive functionality provides an end-to-end approach to enhanced scientific productivity.

The SAVS concept is designed for broad appeal and functionality since its architecture accommodates a general scientific approach to data access, manipulation, analysis and visualization. Every scientific discipline deals with data, either in scalar, vector, or image format. Every discipline needs to develop, test and apply empirical and first-principle model descriptions of cause-effect relationships. Every science needs to deal with the spatial and temporal coordinates of datasets (generally represented by a satellite's orbital elements, payload configuration, and instrument fields-of-view in a space science application); and every scientific discipline requires a suite of mathematical tools to handle data and prepare it for visualization. This is the generic system capability of SAVS, making it applicable to any scientific discipline. In the application described here SAVS is tailored to the needs of the space physics community with user interests in data from past programs (e.g., DE, IMP-8, ISEE, CRRES, etc.), in data provided by recent and imminent satellite operations (e.g., Wind, Polar, Geotail, etc.), and in effective planning of future missions (e.g., TIMED, IMI, and Grand Tour Cluster).

2.3 The Visualization Software

Underlying data handling and analysis in the SAVS system is the AVS visualization software. Designed for a distributed network environment, AVS supports visualization applications running on a single system or across a network of systems. It provides a complete image display capability, including real-time pan and zoom, rotation and transformation, flip-book animation, and support for 8-bit, 24-bit, and floating point images. Imaging filters include operations such as contrast stretching, pseudo-coloring, and histogram balancing, as well as data resizing operations such as interpolation, cropping, and sampling. AVS provides tools for rendering volume data, a real-time isosurface generator, a unique transparent volume renderer which creates real-time semi-transparent images with full rotational and lighting control, and generation of geometric objects such as arbitrary slicing surfaces, dot surfaces, and vector nets. The

software is based on industry-standard graphics, windowing, and programming interfaces including X-Windows and PHIGS+. (For details contact P. Esdale at AVS, Inc., via paule@avs.com.)

With AVS as the pivotal visualization tool, SAVS and its users also benefit from the extensive AVS module library at the International AVS Center (IAC) at the North Carolina Supercomputer Center (see NCSC/IAC in references), which serves as a catalyst for increasing AVS product functionality. As a worldwide repository available at no cost to AVS/SAVS users, the center collects, sorts and distributes user-

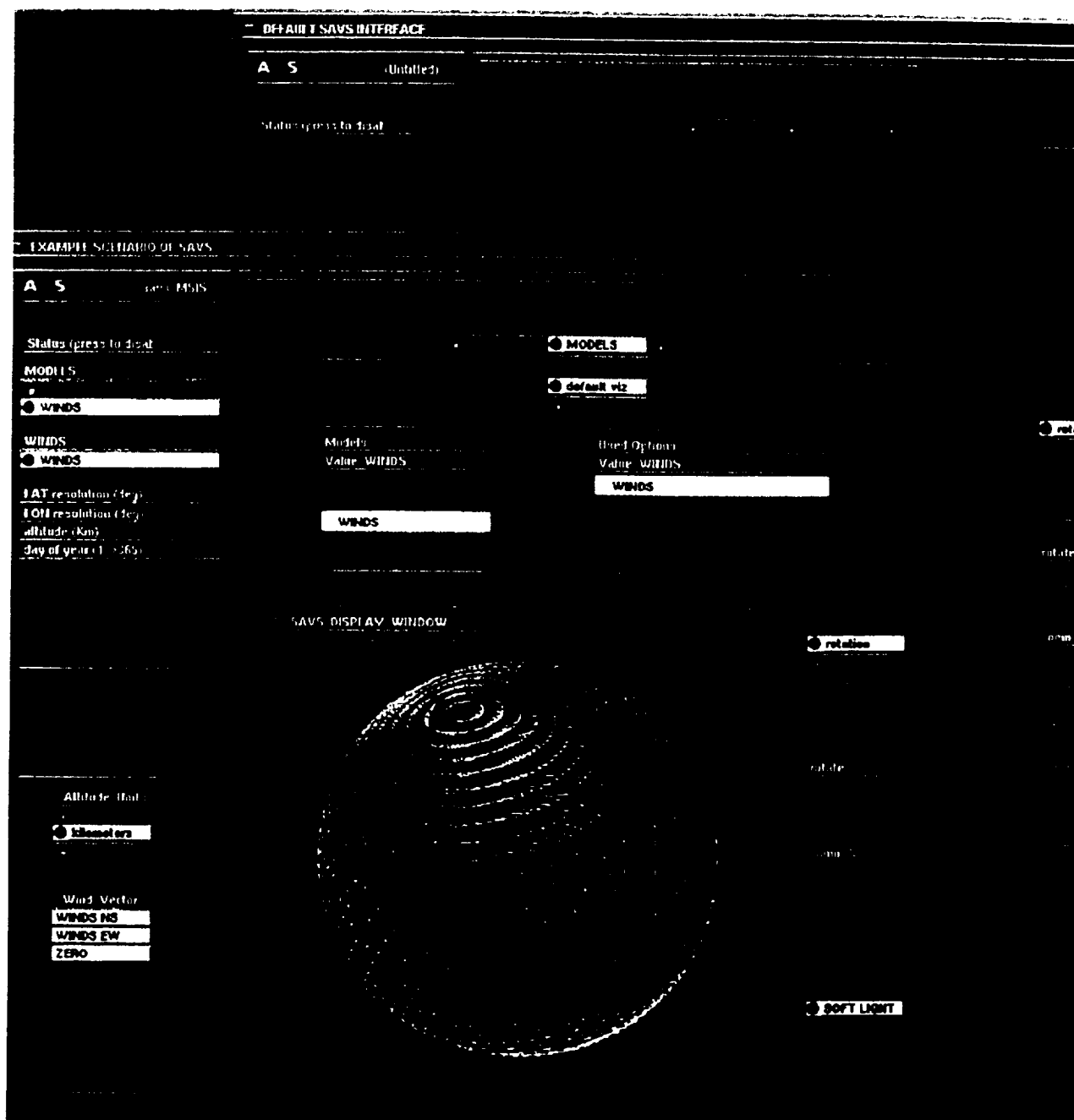


Figure 7. The opening SAVS panel (background plate) gives the user access to the **DATA**, **MODELS** and **ORBITS** concourses, along with push-button utilities for data and model handling and visualization techniques. The front plate is an example of a SAVS application scenario discussed in the text.

contributed public-domain modules. An example of existing aspects of this resource is the availability of 229 Khoros-derived AVS modules [Myerson and Reid, 1993]. Khoros, like AVS, is a data-flow-oriented application, but it makes available to AVS and SAVS its 1-D and 2-D signal processing algorithms, complementing the well-established 3-D volume visualization capabilities in AVS.

The background plate in Figure 7 shows portions of the opening SAVS panels. The top-center of the main panel provides a push-button entry to any of the three SAVS concourses (DATA, MODELS, and ORBITS). The main panel also includes the display window (which is hidden by the front panel) where objects and object overlays are presented and where discrete controls can be used to manipulate the scene with definable increments of rotation, translation, magnification, and reduction.

2.4 The *Push-Button* Front End Architecture

The pull-down menu bar (at the top of the main panel — *controls, filters, etc.*) provides access to a wide spectrum of functions: everything from general application needs to math and visualization tools for multi-variant/multi-dimensional data sets.

The applications panel (vertical panel on left) provides all parameter controls for each model run, for math tools used, and for the visualization process exercised. Behind these controls and most SAVS buttons are tailored networks transparent to the user, enabling him to easily assemble the necessary components for problem analysis without knowledge of the AVS-network building requirements faced by non-SAVS users.

The front panel in Figure 7 illustrates the staging process in a user's entry and exercise of the MODELS concourse. Selection of MODELS presents a choice browser at the top left of the display window which allows the user to search through and select any number of models, including: 1) those in the SAVS library, 2) those which are user-unique and implemented within SAVS by built-in hooks, or 3) those accessed through remote procedure calls. In the current SAVS library are:

- the Feldstein auroral oval model [Holtzworth and Meng, 1975]],
- the International Geomagnetic Reference Field [IGRF, Peddie, 1982],
- the International Reference Ionosphere [IRI, Rawer and Ramanamurty, 1985],
- the MSIS model representation of thermospheric densities and composition [Hedin, 1983],
- the HWM WIND model of thermospheric winds [Hedin, 1991],
- files of MHD model runs of the magnetosphere [Fedder and Lyon, 1987], and
- the Tsyanenko magnetospheric B-field model.

Selecting WIND (as an example of a SAVS application scenario) presents the user with: (1) an automatic visualization default selection of input parameters and a corresponding default visualization format (see pull-down radio buttons under the *MODELS* concourse button in the front panel of Figure 7), (2) the default object in the display window, and (3)

the control panel on the left which allows the user to tune the model and its image according to all its driving variables. For WIND the input variables include dial controls for universal time (UT), the activity index (Ap), and the 10.7 cm solar flux (F10.7). The control panel also offers keyboard controls for selection of computational resolution, altitude for a surface display of the thermospheric wind vectors, day-of-the-year, and scaling of N-S and E-W components of the velocity vectors. The user can pan, zoom, or rotate the image, exercise a parameter study by *instantaneously* changing the input conditions and observing the results, or begin to overlay data or other models (e.g., a color rendering of O₂ and N₂ densities from MSIS, the boundaries of the auroral oval according to Feldstein (with controls over UT and Kp), the IGRF to study meridionally- or zonally-coupled regions, or data for model-measurement comparisons.) An illustration of multiple overlays will be taken up in the discussion of Figure 9.

In the case of model-measurement comparisons, SAVS has a specialized module that projects the results of any three-dimensional model onto any 1-D line [Szuszczewicz, et al., 1993], a requirement for comparisons of along-track satellite observations with large-scale three-dimensional model predictions. Under the ORBITS concourse the user can call up any predefined mission orbit or an orbit file defined by the theoretical CADRE-3 satellite ephemeris package [Galperin, 1990] in the SAVS ORBITS concourse. The user can then implement the 3-D to 1-D interpolation module by activating the *3-D to 1-D* item in the *filters* submenu of the menu bar.

2.5 Data Access

The pivotal point of any analysis effort involves the data, with the SAVS architecture making available the tools to browse, access, and read data from local and remote repositories. While data being accessed by NASA researchers exist in an exceedingly wide range of formats and organizational levels, HDF (hierarchical data format), CDF (common data format), and netCDF are developing broad appeal. HDF, for example, is the baseline standard for EOSDIS data product generation, archival activities, ingest, and distributions. And CDF has been adapted by ISTP for many of its data products. With this perspective, we are providing HDF, CDF and netCDF readers, while including a set of hooks so that users can include readers for non-standard data formats.

To access data from national and international sites, we have integrated prototype browsing and remote procedure calls into SAVS via adaptations of a menu-oriented database front-end. Users, with widely varying applications, can therefore act as their own data retrieval service, thus relieving a long-standing bottleneck connected with poor user access to the data and dependence on data centers for satisfying their requests. While this element of SAVS is still under development, a prototype system has been implemented based on the relational database management techniques employed in SAIC's CenterView [Brennan, 1987]. CenterView itself has a history of more than ten years of development efforts and is routinely used as the hub of a world-wide network which acquires, stores, and analyzes an active real-time global seismic database.

We illustrate the ease and utility of this SAVS element of remote data access, which we call SAVS-View, in an application that can be thought of as generic to interplanetary and magnetospheric physics. Working with the National Space Science Data Center (NSSDC) at the Goddard Space Flight Center we set up a remote library of data and developed an index system and a corresponding set of keys which would allow a user to browse the index catalogs and retrieve the desired subset of data from the library. This

browse-and-retrieve activity involved a number of successful tests and demonstrations, one of which employed an IBM RS/6000 at the University of Colorado linked with an Oracle-based network server at NSSDC.

The background panel in the upper plate of Figure 8 shows the set of keys that our test *user* activated in establishing a query of available data in our library. The panel in the foreground shows the resulting list of data which matched his index of keys after having launched the query. In this case the user was interested in solar wind input to the magnetosphere and wanted to build an empirical picture of that energy using available data and the solar wind energy formalism represented by the epsilon parameter [Perrault and Akasofu, 1978; Vasyliunas et al., 1982] which has the form

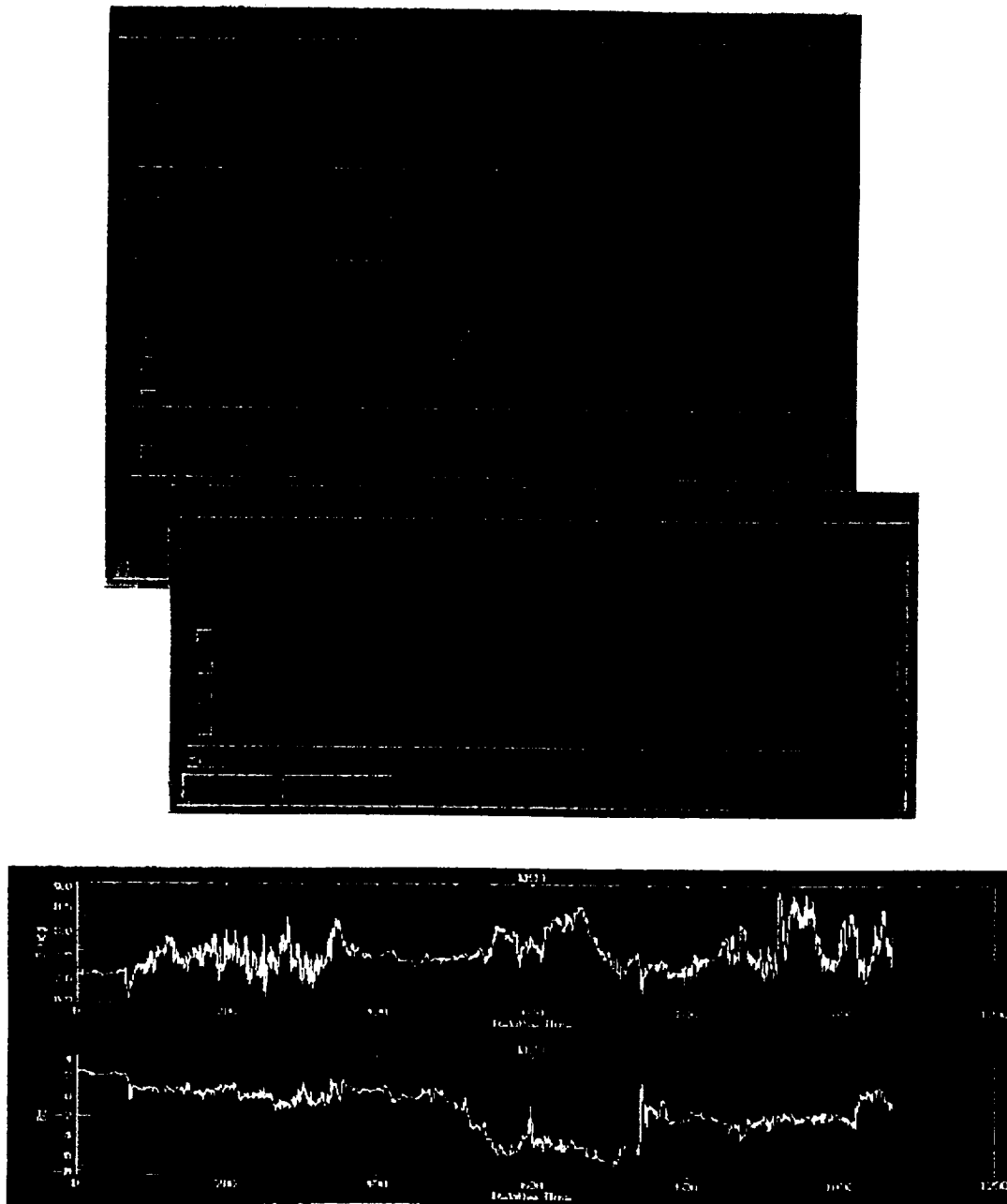


Figure 8. The upper plate shows the set of keys that a test user activated in establishing a query of available data at a remote site. The bottom panel shows a subset of the data acquired, manipulated and displayed within the SAVS system.

$\epsilon = l_0^2 v B^2 \sin^4(\theta/2)$. In this relationship v is the solar wind velocity, B is the value of the interplanetary magnetic field, θ is the polar angle of the component of the IMF normal to the Sun-Earth line and measured from the northward geomagnetic axis, and l_0 is a constant. In order of selection, the user struck the *data* key, and then the *subcategory* key which allowed his choice or choices of measurements. Here, he selected the total B-field, the solar wind velocity and density, and the B_z -component of the IMF as an indicator for θ . His data source of interest was IMP-8, so he keyed Missions and selected IMP-8, and then launched the query using the pull-down query menu at the top of the panel. The entire effort, including the time for SAVS-View to find and list the available data, required approximately two minutes. The transfer of those files and the presentation of the B and B_z parameters using the SAVS 2-D stack plot capabilities are presented in the lower plate in Figure 8. The formatting of the data plots took advantage of SAVS tools that allowed the selection of any parameter in the file for the abscissa and the ordinate. SAVS also allows truncation of the individual plots as well as their extensions along any user-selected domain of the independent variable.

In the development of data access tools, SAVS activities also included the implementation of a SAVS utility to access the Logical Library System (LLS) developed at the NASA Goddard Space Flight Center [Jacobs, 1993]. The LLS, as the name suggests, has the functionality of a library, with programs, missions, events, *etc.* catalogued within the framework of library shelves (*e.g.*, magnetospheric physics), books (*e.g.*, the DE and IMP-8 missions), chapters (*e.g.*, mission years or event categories), and pages (*e.g.*, specific magnetic storms). This system takes advantage of the GSFC/NSSDC Master Directory and will eventually make data at the center available on-line to SAVS calls.

2.6 Visualization Applications to Mission Scenarios

We demonstrate the interactivity and extensibility of the SAVS system with an application scenario that can be considered generic to any mission planning exercise or to an activity involving data reduction and analysis that requires: (i) a number of models, (ii) ground-based and spaceborne data, (iii) *in situ* and remote sensing techniques, (iv) perspectives on flux tube-coupled domains, and (v) co-registration of satellite-borne and ground-based diagnostics.

A strawman scenario for a given mission is presented in Figure 9. The scenario has counterparts in the CRRES, TIMED, or ISTP missions, requiring the coordination of satellite passes with ground-based optical, ionosonde, and radar diagnostics. The MODELS concourse was used in generating the object in the left panel in Figure 9 by overlaying: (i) thermospheric winds (blue vectors) and oxygen densities (global color-coded surface) at 350 km using the SAVS library models WIND and MSIS, (ii) the auroral oval boundaries from the Feldstein model, (iii) altitude cut-planes of ionospheric densities from the IRI in the latitude and longitude planes intersecting the coordinates of the Arecibo Observatory, and (iv) geomagnetic field lines (in white) from the IGRF. The satellite trajectory was extracted from the ORBITS concourse using the CADRE-3 file.

With push-button and dial controls the user can:

- (a) adjust the auroral oval for any magnetic activity index or universal time,

- (b) change the heights of the thermospheric wind and density calculations,
- (c) vary the activity index A_p , the solar F10.7 flux, and the day-of-the year input drivers for WIND and MSIS,
- (d) alter the year for the IGRF,
- (e) tune the month and sunspot number for the IRI, and
- (f) change the coordinates for the intersecting cut-planes (with interest, for example, in high-latitude passes in conjunction with Sondrestrom radar operations in Greenland). The cut-planes could also be in magnetic E-W and N-S directions instead of being aligned with the geographic contours of latitude and longitude at the Arecibo Observatory.

From the CADRE-3 orbit specifications the user can develop stack plots of any number of relevant parameters vs UT (or vs latitude, longitude, MLT (Magnetic Local Time), LT or vs any other parameter). Parameters available to the user include: (1) SZA (Solar Zenith Angle) and L-shell value, (2) point of conjugacy, (3) spacecraft velocity vector relative to the ambient B-field, (4) height of the terminator relative to the satellite track, (5) B-field values or values of its E-W and N-S components, *etc.*

The user can also call-up the 3-D to 1-D module to interpolate any and all model results along any segment of the satellite track and through the Data concourse compare the results with local or remote databases of along-track measurements.



For co-registration with ground-based diagnostics, the right panel of Figure 9 presents a zoomed-in perspective on the Arecibo site that includes the projection of the ground-based HF radar beam up to an altitude of 400 km. That projection is developed and displayed using the SAVS *field-of-view* module that is controllable by the user to represent any and all ground-based remote sensing diagnostics. The user need only input the coordinates of the ground site, the azimuth and elevation of the instrument's line-of-sight, and its beam-width (or acceptance) half-angles in the N-S and E-W directions.

Inspection of the right-hand panel of Figure 9 shows that the satellite (the black sphere) at the given UT in its trajectory is not on a fieldline that connects to the region diagnosed by the ground-based radar. With animation and pause controls, the user can determine the exact time of fieldline-coupling and begin a number of SAVS-controlled operations. For example, the 3-D to 1-D module can be used to interpolate ionospheric and thermospheric model densities, temperatures, and composition onto the connecting fieldline, and the model data used as inputs exercised to calculate flux tube-integrated conductivities from the satellite to the region covered by the ground-based diagnostics. The module could also be used to calculate integrated densities or emissions along the field-of-view of the ground-based sensor. The field-of-view module can also be attached to the spacecraft to project and visualize the field-of view of an on-board remote sensing device. In this case the animation and visualization product could focus on the determination of the time of intersection of the fields-of-view of the satellite and ground-based sensors. The scenarios are unlimited. Models can be tuned and retuned, and results visualized in simple stack plots of along-track correlations for comparisons with data. Visualizations can include constant altitude surfaces of any geophysical

parameters (e.g., N_e , T_e , N_n , T_n and ionospheric-thermospheric constituent densities O^+ , N_2^+ , ..., O , N_2 , etc., as well as thermospheric wind vectors and their meridional and zonal components, etc.) Instead of constant altitude representations, the user can choose altitude cut-planes, isosurfaces, or isocontours. The proper visualization product is only determined by the perceptive and inquisitive scientific mind, with the freedom of the push-button operations of the SAVS system making the choices immediately available.

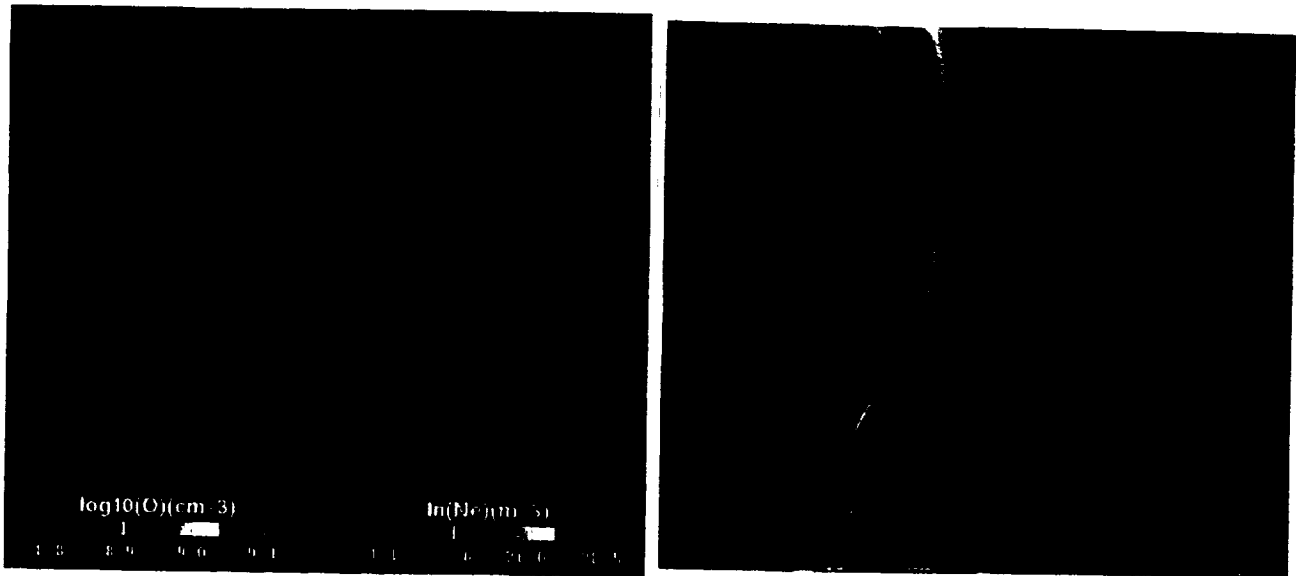


Figure 9. A visualization application of SAVS to a mission scenario with counterparts in the CRRES, TIMED, and ISTP missions. The left panel provides a global view, while the right panel focuses on the conjunction between the satellite pass and the radar field-of-view at the Arecibo Observatory.

2.7 Summary Comments on the Overall System

As described, the SAVS architecture accommodates a generic scientific approach to data access, analysis, and visualization with tools that provide on-line mathematical analysis (e.g., averaging, interpolating, spectral analyses through FFTs, etc.) along with one, two, and three-dimensional displays that can include animation. Data displays and model results can be rendered in an array of 2-D stack plots with the user exercising control of abscissa and ordinate selection from the full set of parameters in the data file. The observations and the model results can be rendered in color codes, isolines, isosurfaces, cut-planes, and texture maps, with controlled lighting and transparency. The 3-D to 1-D interpolation modules can be used for along-track model-measurement comparisons, field-line-integrated calculations, or intensity calculations along the line-of-sight of remote sensing devices. And the field-of-view module can be used to establish regions of co-registration involving multi-satellite missions and ground-based observatories.

The current system has been developed and tested using a relatively large number of models covering magnetospheric, ionospheric, and thermospheric physics [see e.g., Szuszczewicz et al, 1993 and Szuszczewicz, 1994]. And tests involving the data readers and local and remote data access tools have included the CRRES, IMP-8, DE, and the ISEE missions.

The push-button architecture is intended to service user needs focused on science, and not graphics network building and programming. The complexities of the tools are in the background as specialized and tailored modules within the SAVS system framework. The expert user, however, can still access the foundations of AVS and build his or her own graphics network.

While SAVS includes a library of models and data readers, it also provides the hooks and menu-driven recipes for user-unique formats and models in order to guarantee flexibility and extensibility in the overall system. The integrated system is generic in its capabilities with natural applications to ISTP, TIMED, EOS, and TRMM to name only a few of the near-term large-scale science programs at NASA.

3. GENERAL INTERFACE FUNCTIONALITIES

When the SAVS interface is activated, the application panel (narrow vertical panel to the left of the screen in Figure 7), the pull-down menu bar (short wide panel at the top of the screen), and the main SAVS panel (remaining larger panel taking up most of the screen) are displayed. The Network Editor is initiated in a closed (hidden) mode.

The application panel:

The application panel houses all parameter controls for each data reader activated, model run, math tool used, and visualization process exercised.

The pull-down menu bar (*i.e.*, the menu bar):

The pull-down menus control a wide spectrum of functions ranging from general application needs to math and visualization tools for multi-variant/multi-dimensional data sets. For more information, see section 3.3.

The main SAVS panel:

The main panel is partitioned into halves: the *concourse* (the top section) and the *display window* (the bottom section). The *concourse* section controls data retrieval/deletion for SAVS (see section 3.1), while the *display window* section defines and controls visual manipulations (see section 3.2). The controls in each section are independent. Thus, changing the controls in one section will not affect the configuration of the other section.

Brief explanation of an *active data set*. When describing the widgets, panels, and functionality of SAVS in the following text we will be referring to data sets and the *active* data set. A brief explanation of a data set and the active data set follows.

A data set has different meaning for each *concourse* mode. For the *DATA* *concourse* mode, a data set is an ASCII or binary file selected to be read into the SAVS system from the *Data File Browser*. For the *MODELS* *concourse* mode, a data set is a model loaded into the SAVS system selected from the *MODELS* browser. Finally, for the *ORBITS* *concourse* mode, a data set is a 1D set of points read into the SAVS system from an ordinary file browser (for a brief explanation and location of these browsers, see section 3.1).

The *active* data set is the data set that is currently being affected by the SAVS functionality.

3.1 Concourses

The top half of the main panel is known as the *concourse*. It is in this area that the user retrieves/deletes data for SAVS. The user enters the *concourse* in one of three different modes: *DATA*, *MODELS*, or *ORBITS* (for information on each *concourse* mode, see the appropriate section: 3.1.1, 3.1.2, or 3.1.3).

Brief explanation of *concourse* widgets. Not all widgets in the *concourse* section are visible at all times. For instance, only the *DATA*, *MODELS*, and *ORBITS* *concourse* options, the *GLOBE* button, the *Used Options* browser, and *CLEAR OPTION* button are visible at startup. Other widgets are hidden and shown depending on the *concourse* mode the user chooses. In the following descriptions, it is important to note the

dependency of the widget's visibility within the current concourse mode (for example, the *Data File Browser* is only visible when the *DATA* concourse mode is active).

Concourse Widgets

Name	Type	Functionality
DATA, MODEL, and ORBITS	Buttons	By selecting one of these three choices, the user enters that concourse mode and all options chosen from the menu bar act on the current field of that concourse
Data File	Browser	Initially hidden, this browser appears only when the <i>DATA</i> concourse is active. This scrollable browser allows the user to browse and access local data sets.
FTP ACCESS	Button	Initially hidden, this button appears only when the <i>DATA</i> concourse is active. When depressed, <i>FTP ACCESS</i> opens a window that allows the user <i>ftp</i> access to any machine (as long as the server machine name or IP address is known). The retrieved files are placed in the <i>ftp_dir</i> directory defined in <i>savs.scr</i> .
DATA LIBRARY	Button	Initially hidden, this button appears only when the <i>DATA</i> concourse is active. When depressed, <i>DATA LIBRARY</i> activates a remote data browsing and retrieval tool which accesses the Space Science Data Digital Library (SSDdlib) located at Goddard Space Flight Center - Greenbelt Md.
Models	Browser	Initially hidden, this browser appears only when the <i>MODEL</i> concourse is active. This browser contains a listing of models that can be accessed, manipulated, and displayed.
GLOBE	Button	When turned <i>on</i> , a spherical representation of the Earth is shown in the display window. In turn, when turned <i>off</i> , the Earth object is cleared from the display window.

Concourse Widgets (Continued)

Name	Type	Functionality
CLEAR OPTION	Button	Controls the elimination of the highlighted option from the <i>Used Options</i> list, the option's controlling parameters from the application panel, and any objects in the display window created by that option. This process is transparent to the user. For further explanation, see "Purpose and functionality of the <i>CLEAR OPTION</i> - button" below.
Used Options	Browser	Displays a running list of the current concourse's data set(s) and menu bar options added for those data set(s). For further explanation, see "Purpose and functionality of the <i>Used Options</i> browser" below.

Purpose and functionality of the *clear option* button. The *CLEAR OPTION* button takes on different roles, depending on the highlighted item in the *Used Options* browser.

If the highlighted item in the *Used Options* browser is a data set, then by depressing the *CLEAR OPTION* button, SAVS does the following:

- 1) Clears the highlighted data set and all complementing used-menu-bar options from the *Used Options* browser,
- 2) Clears the instance of the highlighted data set and all complementing used-menu-bar options from the Network Editor workspace (transparent to the user),
- 3) Deletes all objects made by the highlighted data set and all complementing used-menu-bar options from the display window, and
- 4) Makes the next data set of the concourse the *active* data set. If the current concourse has no other data sets, SAVS changes to the concourse with the previous *active* data set (this also means the *Used Options* browser list for the new concourse is displayed). The new *active* data set controls appear in the application panel. If there is no previous *active* data set, SAVS takes on the appearance of startup mode.

If the highlighted item in the *Used Options* browser is a used-menu-bar option (*i.e.* starts with -->), then by depressing the *CLEAR OPTION* button, SAVS does the following:

- 1) clears the highlighted item from the *Used Options* browser,
- 2) clears the instance of the highlighted item from the Network Editor workspace (transparent to the user),

- 3) reruns the *active* data set with it's new list of complementing used menu bar options, and
- 4) highlights the *active* data set in the *Used Options* browser and shows its controls in the application panel.

Purpose and functionality of the *Used Options* browser. The *Used Options* browser displays a running list of data set(s) and selected menu bar options for the current concourse mode. All three lists (one for each concourse mode) are independent, and only one list appears in the browser at a time. To change the list appearing in the *Used Options* browser, simply change to the appropriate concourse mode.

There are two types of *Used Options* browser entries: data sets and menu bar options. All menu bar option entries, in the *Used Option* browser list, are prefixed with an arrow symbol (-->). The data sets are not.

By choosing (highlighting) an item from the *Used Options* browser list, the user has several options:

- 1) change the parameters (if any) in the application panel,
- 2) add a menu bar option to the existing string of used options,
- 3) strike *CLEAR OPTIONS*, or
- 4) do nothing.

3.1.1 The *DATA* Concourse Mode

Switching into the *DATA* concourse mode. When accessing the *DATA* concourse mode (for the first time or returning to it from another concourse mode) several things will automatically occur:

- 1) The *FTP ACCESS* and *DATA LIBRARY* buttons will appear under the *DATA* concourse mode button.
- 2) The *Used Options* browser lists all data sets read in and menu bar options used for each data set. The *Used Options* list highlights the first data set (*active* data set).
- 3) The *Data File Browser* will appear between the concourse mode buttons and the display window and highlight the *active* data set, and
- 4) The most recently accessed data-set controls (if any) appear in the application panel.

Reading data into SAVS in the *DATA* concourse mode. SAVS is equipped with several data format readers. The execution of the proper reader is transparent to the user. SAVS identifies the file extension and calls upon the correct reader. For example, if TEST12.cdf were selected from the *Data File Browser*, SAVS would identify TEST12.cdf

as a CDF file and read it with a CDF reader. The following table shows the data formats supported by SAVS and the file extensions needed to identify those formats.

FORMAT	FILE EXTENSION
AVS	fld
CDF	cdf
net CDF	nc
HDF	hdf
SAI image	sai
SUNDIAL IPS	ips

Once the data is read into SAVS, the data's attributes are noted and the unusable menu bar options are disabled (turned off).

3.1.2 The *MODELS* Concourse Mode

Switching into the *MODELS* concourse mode. When accessing the *MODELS* concourse mode (for the first time or returning to it from another concourse mode) several things will automatically occur:

- 1) The *MODELS* browser will appear between the concourse mode buttons and the display window and the most recently accessed model (*active* data set) will be highlighted;
- 2) The *Used Options* browser lists all accessed models and menu bar options used for those models. The *Used Options* list highlights the most recently accessed model (*active* data set); and
- 3) The most recently accessed model's controls (if any) appear in the application panel.

Adding one's own model to the *MODELS* browser. There are several steps to taking a standalone code and placing it into the *MODELS* browser.

- 1) In AVS terms, convert the code into an AVS module.

This is done outside the AVS environment altogether. The user can write an AVS Module by one of the following means: 1) using an existing model module as a template; 2) using the model module outline located in \$(savs_path)/modules/Source/ MODEL_ MODS; or 3) using a combination of the two.

NOTE: The module can be written in FORTRAN or C.

The original model source must be embodied in the model module outline. This means that if *MODEL.c* is used as an outline, the original source must be edited into the *MODEL_compute* routine after the *START MODEL CODE* comment. Likewise, if *MODEL.f.f* is used, the

original source must be written into the *MODEL_compute2* routine after the *START MODEL CODE* comment.

NOTE: In either case, model input parameters are declared in the *MODEL_desc* routine and passed through to the *MODEL_compute* routine (and further passed on to the *MODEL_compute2* routine in the FORTRAN case). For further details see the model module source.

Compile and link the new model module using the appropriate makefile. Place the executable in $\$(savs_path)/modules$.

- 2) Design and save a default network with this module.

This is done in the Network Editor inside the AVS environment (not necessarily from within the SAVS architecture). Save the network into the file $\$(savs_path)/networks/savs_MODEL.net$ (where MODEL is the name of the model as it will appear in the model choice browser).

- 3) Alias a module as *mod_fld* in the saved network in order to supply data to the SAVS environment.

Decide which module, in the saved network, could be described as the main module (usually the model module). The user can alias this module by editing the network (as a normal file, outside of the AVS environment), and adding *-alias mod_fld* into the line that has the desired module. Use $\$(savs_path)/networks/savs_IRI.net$ as an example.

- 4) Add new model module name to the SAVS model list.

This is done by editing $\$(savs_path)/modules/Source/iface.c$. In the interface source, add the model name to the *model_choices* list. Compile *iface.c* by typing *make iface*, and the next instance of the SAVS architecture will include the new model listing.

Execution of a model from the *MODELS* browser. Select a model from the *MODELS* browser. This action causes SAVS to load the model network and display the model parameters in the application panel.

NOTE: When selecting a model from the model choice browser, SAVS looks for the corresponding network to load. For example, if model *foo* were selected from the model choice browser, SAVS would search for and load $\$(savs_path)/networks/savs_foo.net$.

Once the model is read into SAVS, the model's attributes are noted and the unusable menu bar options are disabled (turned off).

3.1.3 The *Orbit* Concourse Mode

Switching into the *Orbit* concourse mode. When accessing the *ORBIT* concourse mode (for the first time or returning to it from another concourse mode) several things will automatically occur:

- 1) A standard file browser will appear between the concourse mode buttons and the display window and the most recently accessed orbit (*active* data set) will be highlighted;
- 2) The *Used Options* browser lists the current orbit and menu bar options used for that orbit. The *Used Options* list also highlights this orbit (*active* data set), and
- 3) The controls (if any) of the current orbit appear in the application panel.

3.2 Display Window

All but two of the widgets to the right of the display window control the *top* object (the compilation of all the separate objects; *i.e.*, the complete scene in the display window). The only exceptions are *CENTER* and *EDIT OBJECT*. These two widgets perform their functions on the current object (the highlighted object in the list produced by depressing the *LIST OBJECTS* widget).

The following list provides a brief explanation of the widgets in the display window section of SAVS.

Display Window Widgets

Name	Type	Functionality
Rotation and Translation	Buttons	In <i>rotation</i> mode, the <i>UP</i> , <i>DOWN</i> , <i>RIGHT</i> , and <i>LEFT</i> buttons rotate the object in the display window by discrete increments defined by the user in the <i>rotate <</i> type-in. While in <i>translation</i> mode, the <i>UP</i> , <i>DOWN</i> , <i>RIGHT</i> , and <i>LEFT</i> buttons translate the object in the display window by discrete increments defined by the user shown in the <i>translate</i> type-in.
Rotate <	Type-in	This editable type-in value identifies the user-defined angular increment for rotating the object
Translate	Type-in	This editable type-in value identifies the user-defined increment for translating the object
UP	Button	Rotates/translates object up in the display window
DOWN	Button	Rotates/translates object down in the display window

Display Window Widgets (Continued)

Name	Type	Functionality
LEFT	Button	Rotates/translates object left in the display window
RIGHT	Button	Rotates/translates object right in the display window
zoom %	Type-in	This editable type-in value defines the discrete amount to zoom in or out on the object (in percentage of the current scene)
ZOOM IN	Button	Magnifies the object in the display window the percentage shown in <i>zoom %</i>
ZOOM OUT	Button	Shrinks the object in the display window the percentage shown in <i>zoom %</i>
RESET	Button	Resets the scene in the display window to zoom in on the origin looking down the Z-axis at the XY-plane
NORMALIZE	Button	Normalizes the scene to fit the display window
CENTER	Button	Sets the center of rotation for the top object to be the center of the highlighted object in the list shown by the <i>LIST OBJECTS</i> button
OMNI LIGHT and SOFT LIGHT	Buttons	In <i>OMNI LIGHT</i> mode, there is no directed lighting in the display window, lighting each object of the scene equally. In <i>SOFT LIGHT</i> mode, a bi-directional light is shone in the display window.
LIST OBJECTS	Button	Displays a scrollable list of objects included in the display window
EDIT OBJECT	Button	Pops up a window which controls various user-definable surface properties of the object chosen in list shown by the <i>LIST OBJECTS</i> button

3.3 Pull Down Menu (i.e., Menu Bar)

Introduction to the menu bar. The menu bar contains four pull-down menus: *CONTROLS*, *GENERAL*, *FILTERS*, and *VIZ TOOLS*.

These options (except those contained in *CONTROLS*) call upon, assemble, and merge single modules or partial networks (both AVS and SAVS-based) with the existing network configuration in order to manipulate data or model output and generate resultant objects for analysis in the display window. This construction is transparent to the user.

Valid menu bar options are determined by scanning the attributes of the *active* data set (for an explanation of *active* data set, see Section 3). For example, if the *active* data set is a scalar field, a vector representation of this scalar field is not valid, and therefore will be a disabled option in the *VIZ TOOLS* pull-down menu. The scanning of the *active* data set and validation of usable menu bar options are completely transparent to the user.

Menu Bar Options.

Options under the <i>CONTROLS</i> pull-down menu:	
Help	Displays on-line help (not implemented)
Display Network Editor	Displays AVS network editor panel and current network
Quit	Closes SAVS interface and exits AVS
Substitute input points	Brings up a choice browser containing a list of all data sets currently in use. SAVS feeds the coordinates of the user-selected data set from this list into the active data set
Add input points	Brings up a choice browser containing a list of all data sets currently in use. SAVS adds the user-selected data set, from this list, to the active data set (<i>i.e.</i> , add another data set to the graph active data set)
Options under the <i>GENERAL</i> pull-down menu:	
View Data	Displays the attributes, coordinates, and data values of the current field in a scrollable window
Write Data	Writes the current field to an external file in an AVS field format
Statistics	Displays general statistics of current field in the application panel
Make 2D Graph	Displays current field in XY-graphical form in separate window

Menu Bar Options (Continued)

Options under the <i>FILTERS</i> pull-down menu:	
Pick Scalar	Extracts a scalar from the current field
Pick Vector	Extracts a vector from the current field
Set minmax	Allows the current field to autoscale or lets the user hardwire the minimum and maximum data values of the current field
Downsize	Defines a subset of the current field by taking every n-th point
Crop	Defines a subset of the current field by extracting a range of data by means of subsetting array dimensions
LLA to XYZ	Converts lat/lon/alt (<i>i.e.</i> , latitude, longitude, altitude) coordinates to XYZ (a spherical to rectangular coordinate transformation)
Vspher2Vrect	Converts vectors from a lat/lon/alt system to XYZ (a spherical to rectangular coordinate transformation for vectors)
Fill points	Linearly interpolates a user-definable number of points between fixed input field points.
Line integral	Calculates the integrated value of a parameter along a 1-D path.
Field math	Allows the user to perform mathematical operations on several data sets (if need be) by using a simplified FORTRAN-like coding.
Options under the <i>VIZ TOOLS</i> pull-down menu:	
Viewer -> Postscript	Copies the scene in the display window to an external file in a Postscript format
Add Label	Adds a label to the display window
Data Bounds	Displays the bounds (extents) of the current field in the display window

Menu Bar Options (Continued)

Options under the <i>VIZ TOOLS</i> pull-down menu: (Continued)	
Data Probe	Interactively retrieves coordinates and data values from the display window using the mouse to locate a position
Dot Representation	Places a sphere (whose radius is user-defined) that is color-coded in accordance with the data value at each coordinate position
Orthogonal Slice	Displays a plane (defined by holding constant one index of the current field's dimensionality) through the data
Arbitrary Slice	Displays an arbitrary plane through the data
Isosurface	Displays a constant-value surface of the current field in the display window
Colored Isosurface	Displays a constant-value surface of the current field and colors it according to another scalar in the display window
Show Vectors	Displays 3-component vectors in the display window as arrows
3D Line Contour	Displays an individual contour of the current field in the display window
3D to 1D (orth)	Displays a 1D interpolation through a 3D volume in graphical form, where the interpolation path is found by holding 2 of the 3 indices constant
3D to 1D (arb)	Displays a 1D interpolation through a 3D volume in graphical form, where the interpolation path is any 1D AVS field

Functionality of the menu bar. SAVS scans the *active* data set, notes its attributes, and enables the appropriate menu bar options. This scanning, noting, and enabling are transparent to the user. After choosing a data set, the menu bar options available for that data are enabled in the menu bar (for an explanation of an *active* data set and how to initiate one, see "Brief explanation of an *active* data set" in Section 3).

To apply one of the menu bar options:

- 1) push and hold the mouse pointer on one of the four menu bar categories,

- 2) drag the mouse pointer down the menu bar category list until the desired option is highlighted, and
- 3) release the mouse button.

When selected, menu bar options (except for those under *CONTROLS*) will be added to and highlighted in the *active* list shown in the *Used Options* browser (for an explanation of the *Used Options* browser, see "Section 3.1 - Concourses"). Those added to the list have the format:

--> *option_name num*

The --> tells SAVS that this item in the *Used Options* browser is a menu bar option (as opposed to no -->, which means the item is a data set), the *option name* is the chosen menu bar option as it appears in the menu bar, and the *num* is for bookkeeping purposes (this integer increments by one each time a new menu bar option is initiated). For example, if *View Data* were initiated, it would appear in the *Used Options* browser as -->*View Data 5* (or some other bookkeeping integer on the end).

As well as appearing in the *Used Options* browser list, the controlling parameters of the chosen option are shown in the application panel. The user is now free to change the option's parameters and observe the results.

Controlling a menu bar option. To redisplay the parameters of a previously chosen menu bar option, make sure the *Used Options* browser list of the proper concourse mode is displayed. After this is done, highlight the appropriate item in the list. The application panel will display that item's parameters and the user is free to change them.

4. HANDS-ON APPLICATIONS

Throughout the following examples there will exist two modes: an *I want* mode, which will list what the user wants to do (text following *WANT>*), and a *do this* mode, which will tell the user what to do to accomplish the desired task (text following *DO>*). The following examples assume that SAVS has been loaded and initiated.

4.1 Data-Model Comparison: Compare Auroral Image Data and Auroral Oval Model

This application deals with model-measurement comparisons. The data is a DE/SAI auroral oval image acquired remotely from NASA/NSSDC using the LLS. The data file (a single *demonstration* image that is part of the SAI/DE database), however, now resides in the SAVS data concourse. The exercise accesses and reads in the image, displaying a default object. From the models library the Feldstein oval is accessed and superimposed on the image data, with model controls of UT and the activity index Q in the application panel. The final product will look like that which is shown in Figure 10.

WANT> read in auroral image *83131002209.sai*

DO> select *DATA* concourse mode

DO> locate and highlight *83131002209.sai* from "Data File Browser"

WANT> visualize SAI image on globe

DO> select *LLA to XYZ* option from *FILTERS* pull-down menu

DO> choose *earth radii (LLA to XYZ parameter)*

DO> select *Dot Representation* from *VIZ TOOLS* pull-down menu

DO> depress *RESET* on the display window panel

DO> depress *NORMALIZE* on the display window panel

DO> depress *CENTER* on the display window panel

DO> change *Radius Scale (Dot Representation parameter)* to roughly 0.0175

DO> depress *GLOBE* on the main panel (left of concourse modes)

DO> select *OMNI LIGHT* (display widow widget)

WANT> look at southern hemisphere

DO> enter 180 in *rotate <* parameter (display window widget)

DO> depress *UP* (display window widget)

WANT> overlay Auroral Oval model on top of SAI image

DO> select *MODELS* concourse mode

DO> select *Auroral_Oval* from *MODELS* choice browser

WANT> change parameters of Auroral Oval model to better fit SAI image

DO> select *auroral oval* from Application panel

DO> change *q_value* (an *auroral oval* parameter) to a new value

DO> change *UT* (an *auroral oval* parameter) to a new value

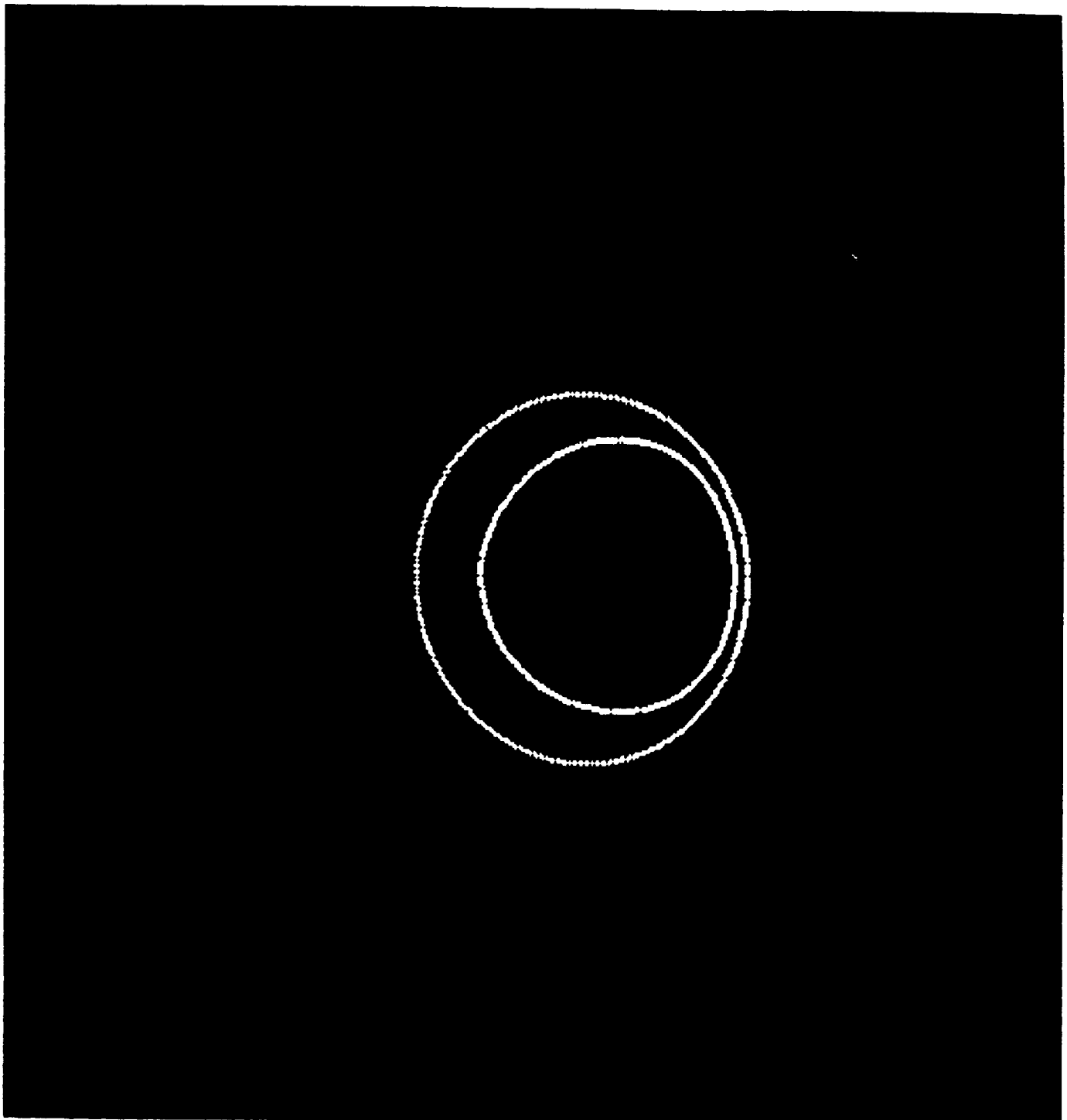


Figure 10. SAVS panel display of a comparison between the Feldstein Oval model and a DE/SAI image of an auroral oval.

4.2 Integrating Models and the Math Module Utility with Stack Plots: Calculations of Field-Line Integrated Conductivities

This application invokes several models from the SAVS library, the CRRES G-9 orbit from the *Orbit* concourse, the 3D-to-1D interpolation module, and the math module. The "problem to be solved" involves the scientific need to calculate field-line integrated conductivities known to control E and F-region coupling mechanisms and overall ionospheric electrodynamics. The equations for the parallel σ_o , Hall σ_1 , and Pederson σ_2 conductivities (Equations 1-3), along with simple formulae for collision frequencies (ν_{in} and ν_{en}), have been entered into the math module using Fortran syntax. Calculation of those conductivities requires, electron and neutral densities, ion composition, magnetic field values (for the electron and ion cyclotron frequencies,

$$\sigma_o = ne^2 \left(\frac{1}{m\nu_{en}} + \frac{1}{m_i\nu_{in}} \right) \quad (1)$$

$$\sigma_1 = ne^2 \left(\frac{\nu_{in}}{m_i(\omega_{ce}^2 + \nu_{in}^2)} + \frac{\nu_{en}}{m_e(\omega_{ce}^2 + \nu_{en}^2)} \right) \quad (2)$$

$$\sigma_2 = ne^2 \left(\frac{\omega_{ce}}{m_e(\omega_{ce}^2 + \nu_{en}^2)} + \frac{\omega_{ci}}{m_i(\omega_{ci}^2 + \nu_{in}^2)} \right) \quad (3)$$

ω_{ce} and ω_{ci} , respectively), and electron temperatures, and all must be functions of time, season, and sunspot number. These values are extracted from the IRI and MSIS models in the SAVS library and a typical path of integration might be the line of site of a radar, or the field line which couples a satellite position with some ground-based diagnostic (e.g., a radar, ionosonde, or Fabry-Perot interferometer). We choose the latter in this illustration, looking to calculate the conductivities from the CRRES perigee pass (at 441 km) to the 100 km altitude point above the Arecibo observatory. The IGRF and the satellite position define the line of integration with the results in stack plot format shown in Figure 11.

WANT> read in orbit and identify satellite

DO> select *ORBITS* concourse mode

DO> locate and highlight *G-09all.xdr.fld* from *Read File Browser*

DO> depress *RESET* on the display window panel

DO> depress *NORMALIZE* on the display window panel

DO> depress *CENTER* on the display window panel

DO> change *position #* (*ORBITS* parameter) to 25

DO> note position of satellite from applications panel

DO> change *Radius* (*ORBITS* parameter) to .05

WANT> run B-field line through specific point along orbit

DO> select *MODELS* concourse mode
DO> select *Bfield_lines* from *MODELS* browser
DO> select *Starter Points* from applications panel
DO> select *POINT* (*Starter Points* parameter)
DO> set *ALT min (Km)* = altitude of satellite (*Starter Points* parameter)
DO> set *ALT max (Km)* = altitude of satellite (*Starter Points* parameter)
DO> set *LATITUDE* = latitude of satellite (*Starter Points* parameter)
DO> set *LONGITUDE* = longitude of satellite (*Starter Points* parameter)

WANT> crop B-field line from orbit to surface of Earth

DO> select *Crop* option from *FILTERS* pull-down menu
DO> change *Z MAX* (*Crop* parameter) to 400

WANT> fill in B-field segment with more points

DO> select *fill points* from *FILTERS* pull-down menu
DO> set *fill_pts* (*fill points* parameter) to 10

WANT> run MSIS and IRI with B-field segment as *seed* points

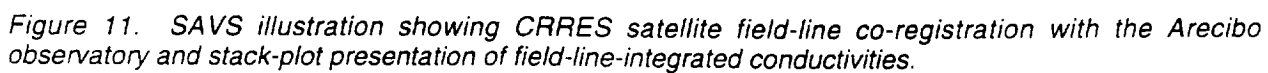
DO> select *MODELS* concourse mode
DO> select *MSIS* from *MODELS* browser
DO> select *substitute input points* from *CONTROLS* pull-down menu
DO> select -->*fill points* from popup *points* browser
DO> strike *RUN* (*MSIS* parameter)
DO> select *IRI* from *MODELS* browser
DO> select *substitute input points* from *CONTROLS* pull-down menu
DO> select -->*fill points* from popup *points* browser
DO> select *LLA to XYZ* option from *FILTERS* pull-down menu
DO> strike *RUN* (*IRI* parameter)
DO> select *IRI* from *Used Options* browser

WANT> calculate conductivities

DO> select *Field Math* from *FILTERS* pull-down menu
DO> select *add input points* from *CONTROLS* pull-down menu
DO> select *MSIS* from popup *points* browser
DO> select *Read Formula* (*Field Math* parameter)
DO> locate and highlight *conductivities.fmla* from popup browser
DO> strike *execute* (*Field Math* parameter)

WANT> plot conductivities

DO> select *Make 2D Graph* from *GENERAL* pull-down menu
DO> select *add input points* from *CONTROLS* pull-down menu
DO> select -->*Field Math* from popup *points* browser
DO> select -->*Field Math* from popup *points* browser
DO> select *PLOT 1* (*Make 2D Graph* parameter)
DO> select *Z coord* from *X Points* (*Make 2D Graph* parameter)
DO> select *parallel* from *Y Points* (*Make 2D Graph* parameter)



DO> select *PLOT 2 (Make 2D Graph parameter)*
DO> select *Z coord* from *X Points (Make 2D Graph parameter)*
DO> select *Pedersen* from *Y Points (Make 2D Graph parameter)*
DO> select *PLOT 3 (Make 2D Graph parameter)*
DO> select *Z coord* from *X Points (Make 2D Graph parameter)*
DO> select *Hall* from *Y Points (Make 2D Graph parameter)*

WANT> integrated conductivity

DO> highlight *Field Math* in *Used Options* browser
DO> select *line integral* from *FILTERS* pull-down menu

WANT> plot integrated conductivity

DO> highlight *Make 2D Graph* in *Used Options* browser
DO> select *add input points* from *CONTROLS* pull-down menu
DO> select -->*line integral* from popup *points* browser
DO> select *PLOT 4 (Make 2D Graph parameter)*
DO> select *Z coord* from *X Points (Make 2D Graph parameter)*
DO> select *DATA* from *Y Points (Make 2D Graph parameter)*

REFERENCES

- Botts, M.E., The State of Scientific Visualization with Regard to the NASA EOS Mission to Planet Earth, Report to NASA Headquarters, July 1992.
- Brennan, M.A., "Center for Seismic Studies Database Structure Version 2.8," Tech. Rpt. C87-04, Teledyne-Geotech, Arlington, VA (September 1987).
- Fedder, J.A. and J.G. Lyon, The solar wind-magnetosphere-ionosphere current-voltage relationship, Geophys. Res. Lett., **14**, 880, 1987.
- Friedhoff, R.M. and W. Benzon, The Second Computer Revolution: Visualization, Harry N. Abrams, Inc., New York, 1989.
- Galperin, Y. "An IBM PC - Compatible Software Package for Satellite Orbit Definition Ephemeris Determination, and Along-Track Geophysical Parameters," (private communication, 1990).
- Foster, et al., Time Dependent Animation Software for Analysis of NCAR Thermosphere/Ionosphere General Circulation Model (TIGCM) Results, and Comparison with Satellite Data, in Visualization Techniques in Space and Atmospheric Sciences, edited by E.P. Szuszczewicz and J. Bredekamp, NASA Headquarters Publication, 1994.
- Gekleman, W., Making Sense from Space-Time Data in Laboratory Experiments on Space Plasma Processes, in Visualization Techniques in Space and Atmospheric Sciences, edited by E.P. Szuszczewicz and J. Bredekamp, NASA Headquarters Publication, 1994.
- Hedin, A.E., M.A. Biondi, R.G. Burnside, G. Hernandez, R.M. Johnson, T.L. Killeen, C. Mazaudier, J.W. Meriwether, J.E. Salah, R.J. Sica, R.W. Smith, N.W. Spencer, V.B. Wickwar, and T.S. Viridi, Revised Global Model of Thermosphere Winds Using Satellite and Ground-Based Observations, J. Geophys. Res., **96**, 7657, 1991.
- Hedin, A.E., "A Revised Thermospheric Model Based on Mass Spectrometer and Incoherent Scatter Data: MSIS-83," J. Geophys. Res. **88**, 10710 (1983).
- Holtzworth, R.H. and C.I. Meng, "Mathematical Representation of the Auroral Oval," Geophys. Res. Lett. **2**, 337 (1975).
- Jacobs, B.E., "Implementation Plan for the Space Science Data Digital Library (SSDdlib)", NASA/GSFC Draft Report (November 16, 1993).
- Keller, P.R., and Mary M. Keller, Visual Clues: Practical Data Visualization, IEEE Computer Society Press, Manning Publ. Col., Greenwich CT, 1993
- Myerson, T. and D. Reid, "New Khoros Modules for AVS," AVS Network News **1**, Is. 3, 6-10 (1993); distributed by the University of Manchester through anonymous FTP to ftp.mcc.ac.uk.

NCSC/IAC, North Carolina Supercomputing Center, Int'l AVS Ctr., 3021 Cornwallis Rd., Research Triangle Park, NC 27709, anonymous FTP to avs.ncsc.org.

Nielson, G.M., B. Shriver, L.J. Rosenblum, editors, Visualization in Scientific Computing, IEEE Computer Society Press, Los Alamitos, CA 1990.

Peddie, N.W., "International Geomagnetic Reference Field: the Third Generation," J. Geomagn. Geoelectr. **34**, 309-326 (1982); for revisions contact the Nat'l. Geophys. Data Ctr., Boulder, CO (303-497-6478).

Perreault, P. and S.-I. Akasofu, "A Study of Geomagnetic Storms," Geophys. J. Roy. Astron. Soc. **54**, 547 (1978).

Ratcliffe, J.A., An Introduction to the Ionosphere and the Magnetosphere, Cambridge University Press, 1992.

Rawer, K., and Y.V. Ramanamurty, "International Reference Ionosphere - Status 1985/86," URSI/COSPAR Workshop Proceedings, Belgium, 28 October 1985, Wheaton & Co., Ltd., UK, 1985.

Szuszczewicz, E.P., A. Mankofsky, P. Blanchard, C. Goodrich, D. McNabb, and D. Kamins, "SAVS: Progress and Plans," in AISRP Workshop III Proceedings, G. Mucklow, editor, Published by NASA Information Systems Branch, Wash. D.C. (1993).

Szuszczewicz, E.P., "Visualization: A Pathway to Enhanced Scientific Productivity in the Expanding Missions of Space and Earth Sciences," in Visualization Techniques in Space and Atmospheric Sciences, E.P. Szuszczewicz and J. Bredekamp editors, NASA Printing Office, (Wash. D.C., 1994).

Twiddy, et al., Visualization of a 3-Dimensional Ozone Data Set: Nimbus-7 SBUV (abstract), EOS, Trans., AGU. **74** (16), Spring Meeting Suppl. 268, 1993.

Upson, C., T. Faulhaber, Jr., D. Kamins, D. Laidlaw, D. Schlegel, J. Vroom, R. Gurwitz, and A.V. Dam, "The Application Visualization System: A Computational Environment for Scientific Visualization," IEEE Computer Graphics and Applications, 30-42 (July 1989).

Vasyliunas, V.M., J.R. Kan, G.L. Siscoe, and S.-I. Akasofu, "Scaling Relationships Governing Magnetospheric Energy Transport", Space Sci. **30**, 359-365 (1982).

APPENDIX -- DATA

AVL Modules

read cdf

NAME

read cdf - read CDF data from a disk file and convert into AVS field format

SUMMARY

Name read cdf

Availability Advanced Visualization Laboratory, U of MD.

Type data

Inputs none

Outputs field

Parameters	Name	Type
	Read CDF File	Browser
	Variable	Browser
	Min,Max Record Dial	
	CDF Info	Boolean
	Single/Multi Records	Radio Buttons
	Output Field	Boolean
	Scalar/N-vector radio button	

DESCRIPTION

The Read CDF module reads variables in the CDF(Common Data Format) and maps them to uniform AVS fields.

PARAMETERS

Read CDF File

A file browser window to specify the name of the CDF file to be read.

Variable

A choice browser to select one of a number of variables in the CDF dataset.

Min/Max Records

Selects record range to be selected from the CDF dataset.

CDF Info

Displays CDF header and Variable attribute information.

Single/Multi Record(s)

Determines whether a single record is output or a range of records is output from the CDF file.

Output Field

If on the CDF data is sent through the output port.

AVS Vector Type

Determines which if any dimension to be mapped as an N-vector in an AVS field.

scalar - Do not map to an N-vector in the AVS field

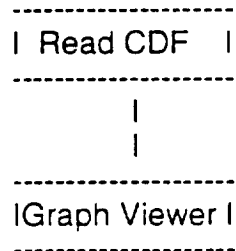
1st-vector - map the first CDF dimension as an N-vector

Last-vector - map the last CDF dimension as an N-vector

RELATED MODULES

The Read CDF module can be used with any module that accepts field type input. As the Read CDF module can output 1D, 2D, and 3D uniform fields the user must take care that the destination modules can accept a similar range of data.

EXAMPLE



Pre Release

AVL

read cdf

NAME

Read netCDF&HDF - input a scientific data set from a netCDF or HDF format file.

SUMMARY

Name Read HDF&netCDF

Type Input

Inputs NONE

Outputs field - field

Parameters

Name	Type	Choices [Default]
filename	browser	[.nc,.hdf]
Selection	radio_buttons	Data,Coord [Data]
Irreg	radio_buttons	uniform,rectilinear,irregular
[uniform]		
Data Set	choice_browser	data sets in file
X	choice_browser	"appropriate" data sets
Y	choice_browser	"appropriate" data sets
Z	choice_browser	"appropriate" data sets
XYZ	radio_buttons	X,Y,Z [X]
Info	radio_buttons	File, DS, None
Attributes	text_browser	File or Data Set attributes
scalar	toggle	[on]
vector_first	toggle	[off]
scale_data	dial	[1.0]
x_scale	dial	[1.0]
y_scale	dial	[1.0]
z_scale	dial	[1.0]

DESCRIPTION

This module reads a data set from either an HDF file or a netCDF file. It assumes the data is a 1, 2, or 3 dimensional array with n vector components. The data must be stored as (l,n) or (n,l), where l can be i, i,j or i,j,k and is the spatial indices. "n" is the vector component index. The data can be a vector or scalar array. Vector data can be read as (n,i,j,...) or (i,j, ...,n) by user selection (vector_first parameter). Data types real, integer, byte (char), double and short are supported. Eight bit integer data is interpreted as byte.

The user can also set whether the output AVS field is uniform, rectilinear, or irregular. If rectilinear or irregular, the user is prompted to select data sets from the file to use as coordinate data. After a data set is selected, clicking the Coord button will expose controls to set the coordinates. The module will then search the file for data sets matching the dimensions of the data set and offer appropriate choices to the

user for the X, Y, and Z coordinates. Note for rectilinear coordinates, sets of coordinates will be limited by the rank of the data set - ie, only X and Y will be available for 2D data, and only X for 1D data.

PARAMETERS

filename (browser) - Input HDF or netCDF file. Only files with .hdf or .nc suffixes appear in browser.

Selection (radio_buttons) - Operation Mode: Data or Coord.

Info (radio_buttons) - Show file, data set (or coordinate data set depending on Selection), or not.

Attributes (text_browser) - Text window with file or data set attributes

Irreg (radio_buttons) - Type of (output) AVS field desired:

uniform, rectilinear or irregular. uniform is the default.

DATA SET - the following parameters are displayed and are effective when Data is selected

Data Set (choice_browser) - Data set within file to be used as the AVS field data. The choices in the browser are set to the data sets in the file after valid HDF or netCDF file is selected.

scalar (toggle) - Treat data set as a scalar array if true; treat as a vector array if false. The default is true.

vector_first (toggle) - If data set is a vector array- scalar is false - assume the vector index is the FIRST of the data set dimensions, ie, ds(n,i,j,...), if true. If false, assume the vector index is the LAST of the data set dimensions, ie, ds(i,j,...,n).

scale_data - normalize (divide) data by this value. The default value is 1.

COORDINATE - the following parameters are displayed and are effective when Coord is selected. A Data Set MUST be selected before these parameters can be used.

XYZ(radio_buttons) - Coordinate [X, Y, or Z] to modify. If the AVS field is rectilinear, the rank of the data determines the selections possible, ie, for 2D data, only X and Y choices are presented.

X (choice_browser) - Data set within file to be used as the X coordinates in the AVS field. Only data sets with rank and dimensions appropriate for the value of Data Set are displayed in the browser.

Y (choice_browser) - Data set within file to be used as the Y coordinates in the AVS field. Only data sets with rank and dimensions appropriate for the value of Data Set are displayed in the browser.

Z (choice_browser) - Data set within file to be used as the Z coordinates in the AVS field. Only data sets with rank and dimensions appropriate for the value of Data Set are displayed in the browser.

x_scale (dial) - normalize (divide) X coordinates by this value. The default value is 1.

y_scale (dial) - normalize (divide) Y coordinates by this value. The default value is 1.

z_scale (dial) - normalize (divide) Z coordinates by this value. The default value is 1.

OUTPUTS

field - AVS field of data from input HDF or netCDF file.

AUTHOR

Advanced Visual Lab
University of Maryland
College Park MD

Release 1

SAVS
Read_netCDF&HDF(6)



NAME

SAI_to_field - linear interpolation of input field

SUMMARY

Name SAI_to_field

Type data input

Inputs NONE

Outputs field 2D 2-space irregular float

Parameters	Name	Type	Default	Min	Max
saifile	browser		.sai		

DESCRIPTION

This module reads an SAI image file (supplied by the DE satellite) and converts it to an AVS field. The file browser only lists directories and files with "sai" file extensions.

PARAMETERS

saifile

This file browser only lists directories and files with the "sai" file extensions

OUTPUTS

Data Field

The value at each coordinate represents the image intensity

Release 1	SAVS	SAI_to_field(6)
-----------	------	-----------------

NAME

IPS_READER - linear interpolation of input field

SUMMARY

Name IPS READER

Type data input

Inputs NONE

Outputs field 1D 3-space 3-vector irregular float

Parameters	Name	Type	Default	Min	Max
	datafile	browser			ips
	UT	float	1.	0.	24.
	day_min	integer	1	1	31
	day_max	integer	1	1	31
	run	oneshot	0	0	1
	month	choice			
	year	choice			
	code	choice			

DESCRIPTION

This module reads an IPS-ionosonde data format (originated by Dr. Phil Wilkinson at the Ionospheric Prediction Service Radio and Space Services, Australia) and converts it to an AVS field. The file browser only lists directories and files with "ips" file extension.

When an IPS file is read, this module scans the file for month, year, and code information. It places this information in radio-button form in a separate window. It is then up to the user to select a month, year, and code to search for.

When a month is selected, the day_min and day_max dials reset themselves to display the maximum daily range. The user is free to specify a range of days to output as well. If a range of days is chosen, this module divides the data daily sum by the number of valid data days and outputs this average (invalid data corresponds to 0.).

IPS files list data on the hour for a given day beginning with hour 1 (not 0) and continuing up to hour 24 (not 23). If the user selects a time less than 1 UT, this module linearly interpolates between the last hour of the previous day and the first hour of the current day to calculate the correct value. If the user selects a time not on the hour (but greater than 1 UT), this module linearly interpolates between the hourly values that straddle it. This module will interpolate over a maximum of 3 hour windows.

When all the parameters are set and the data is ready to be extracted, depress run. At this point data will be sent to the output field if 33% of the selected daily range is

valid (*i.e.*, if 10 of the selected 30 days is valid data, write to output; if only 2 of the selected 15 days is valid data, skip over). A valid data point has three values: the sum of valid data divided by number of valid days, the number of valid days, and the corresponding station number.

Once data has been selected and extracted, this module reads the station number(s) from the data and looks up in "NAMES.STN" the correct coordinate information for that station. Along with the data value, the number of good days and station number are listed as output data values as well.

PARAMETERS

datafile

This file browser only lists directories and files with the "ips" file extension

UT - universal time

day_min - minimum day to average the data

day_max - maximum day to average the data

run - depress to extract the data from the IPS file (once the parameters are set)

month - once a data file is highlighted in the browser, the available months are listed in column form (the user chooses one)

year - once a data file is highlighted in the browser, the available years are listed in column form (the user chooses one)

code - once a data file is highlighted in the browser, the available codes are listed in column form (the user chooses one)

OUTPUTS

Data Field

The output field is a list of points, where each point is comprised of three values at a station location.

The three data values at each point are: the data averaged over the chosen day range for the selected parameters, number of days with valid data, and station number. The coordinates are identified by cross-referencing the station number in NAMES.STN.

Release 1

SAVS

IPS_READER(6)

NAME

Starter_Points - defines "seed" points to feed into other modules

SUMMARY

Name Starter Points

Type data input

Inputs NONE

Outputs field 3-space 0-vector float

Parameters	Name	Type	Default	Min	Max
	alt_min	integer	0	0	10000
	alt_max	integer	5000	0	10000
	alt_step	integer	250	1	10000
	pt_distr	choice	GLOBAL	GLOBAL:SLICE:POINT	
	slice_ch	choice	LAT SLICE:LON SLICE		
	lat	float	0.	-90.	90.
	lon_min	float	0.	0.	360.
	lon_max	float	360.	0.	360.
	lon_step	float	5.	0.	180.
	lon	float	0.	0.	360.
	lat_min	float	-90.	-90.	90.
	lat_max	float	90.	-90.	90.
	lat_step	float	5.	0.	90.
	coord_sys	choice	LAT-LON-ALT LAT-LON-ALT:X-Y-Z		

DESCRIPTION

This module defines seed points to be used as inputs into other modules. The output field is selected by choosing the type of point distribution and detailing the mesh ranges and interval step sizes.

The output field is defined by parameters described in a spherical coordinate system (better understood by most). The output coordinates can be generated for a rectangular coordinate system by selecting X-Y-Z on the coord_sys parameter

The output field has no data, only coordinate information

INPUTS

NONE

OUTPUTS

If user chooses LAT-LON-ALT from coord_sys parameter, the output field will be 3D rectilinear with coordinates in a spherical coordinate frame. Otherwise, if X-Y-Z is chosen, the output field will be 1D irregular with coordinates in a rectangular coordinate frame.

PARAMETERS

alt_min \
alt_max > define altitude mesh of seed points (in units of Km)
alt_step/

pt_distr - define which type of point distribution the user wants

slice_ch - if SLICE is chosen from pt_distr, this parameter appears and allows the user to select the type of slice

lat - define latitude node (deg). visible if SLICE AND LAT SLICE are choosen OR POINT is choosen.

lon_min \
lon_max > define longitude mesh (deg). visible if GLOBAL is choosen OR
lon_step/ SLICE AND LAT SLICE are choosen.

lon - define longitude node (deg). visible if SLICE AND LON SLICE are choosen OR POINT is choosen.

at_min \
lat_max > define latitude mesh (deg). visible if GLOBAL is choosen OR
at_step/ SLICE AND LON SLICE are choosen.

coord_sys - define coordinate system of output field



Release 1

SAVS

Starter_Points(6)

APPENDIX -- MODELS

SAVS Modules

IRI_F(6)

NAME

IRI_F - International Reference Ionospheric model

SUMMARY

Name IRI (F)

Type data input

Inputs field 3-space float

Outputs field 3-space float

Parameters	Name	Type	Default	Min	Max
	run	oneshot	0	0	1
	doutput	choice	Peak	Values	Profile
Values	UT	float	12.	0.	24.
	MONTH	integer	10	1	12
	R	float	145.	0.	145.

DESCRIPTION

```

C*****
C***** INTERNATIONAL REFERENCE IONOSPHERE (IRI). *****
C*****
C***** IRIMP9, MARCH 1987 *****
C***** DRIVER PROGRAM *****
C*****
C*** THIS PROGRAM PRODUCES PROFILES OF ***
C*** ELECTRON DENSITY ***
C*** NORMALISED ELECTRON DENSITY (TO F-MAXIMUM) ***
C*** NEUTRAL TEMPERATURE (CIRA 72) ***
C*** ELECTRON TEMPERATURE ***
C*** ION TEMPERATURE ***
C*** ELECTRON TO ION TEMPERATURE RATIO ***
C*** RELATIVE PRECENTAGE DENSITIES OF THE IONS ***
C*** ATOMIC OXYGEN, HYDROGEN, HELIUM, ***
C*** MOLECULAR OXYGEN AND NITROGEN OXYD (NO+) ***
C*****
C*** FOR SPECIFIED ***
C*** GEOG. LATITUDE AND LONGITUDE ***
C*** ALTITUDE ***
C*** ZURICH SUNSPOT NUMBER (12-MONTH-RUNNING MEAN) ***
C*** MONTH-NUMBER ***
C*** UNIVERSAL TIME ***
C*****
C*** THE ALTITUDE LIMITS ARE: LOWER UPPER ***

```



```

C*** ELECTRON DENSITY    60(DAY)/80 KM    1000 KM    ***
C*** TEMPERATURES      120 KM            3000 KM    ***
C*** ION DENSITIES      100 KM            1000 KM    ***
C*****
C*   AUTHORS OF ORIGINAL IRI STAND ALONE CODE    *
C*   -----*
C*   I PROF. K. RAWER      DR. D. BILITZA      |    *
C*   I HERRENSTR. 43      GSFC/NSSDC CODE 633   |    *
C*   I D-7801 MARCH      GREENBELT MD 20771    |    *
C*   I F.R.G.            USA                    |    *
C*   -----*
C*****
C*****
C***** ALL ANGLES ARE IN DEGREE *****
C***** ALL DENSITIES ARE IN M-3 *****
C***** ALL ALTITUDES ARE IN KM *****
C***** ALL TEMPERATURES ARE IN KELVIN *****
C***** ALL TIMES ARE IN DECIMAL HOURS *****
C*****
C*****
C*** THIS VERSION DIFFERS FROM IRIMP8 BY CONTAINING: ***
C*** - RESTRICTION OF RZ12/COVM TO VALUES BELOW 145/185 ***
C*** - CORRECTED HARMONIZED BENT *****
C*** - NEW ELECTRON TEMPERATURE MODEL *****
C*** - NEWER FOE NIGHT-TIME *****
C*****
C*****

```

To execute this module, the following files must be loaded into a known directory: ccir1c.dat, ccir2c.dat and ccir3c.dat. The path to this known directory must be changed in the module source at the three "open" statements.

INPUTS

Field (field 3-space float; REQUIRED)

The input field defines the seed points for the IRI model. It is at this field's coordinates that the IRI will calculate the desired output. The IRI can accept rectilinear OR irregular point meshes (and output the same) from any source as long as the coordinate information is 3-space in a spherical system (*i.e.*, LAT/LONG/ALT). It is for this reason that the module "Starter Points" is ideal input for the IRI model module.

PARAMETERS

run - depress to execute IRI model and send desired output to output port

doutput - selection to choose between two sets of output vectors. See OUTPUTS for more details

UT - universal time

month - month

R - ZURICH sunspot number (12-month-running mean)

OUTPUTS

Data Field - The output field depends on the doutput parameter. If Profile Values is chosen, the output is 11-vector. Or if Peak Values is chosen, the output is 8-vector. The type of ouput mesh is the same as the input mesh. The output values are given below.

Profile Values

OUTF(1)=NE [CM-3]
OUTF(2)=NE/NMF2 -
OUTF(3)=TN-CIRA [K]
OUTF(4)=TI [K]
OUTF(5)=TE [K]
OUTF(6)=TE/TI -
OUTF(7)=O+
OUTF(8)=H+
OUTF(9)=HE+
OUTF(10)=O2+
OUTF(11)=NO+

Peak Values

OUTF(1)=HMF2 [KM]
OUTF(2)=HMF1 [KM]
OUTF(3)=HME [KM]
OUTF(4)=HMD [KM]
OUTF(5)=NMF2 [CM-3]
OUTF(6)=NMF1 [CM-3]
OUTF(7)=NME [CM-3]
OUTF(8)=NMD [CM-3]

NOTES

To execute this module, the following files must be loaded into a known directory: ccir1c.dat, ccir2c.dat and ccir3c.dat. The path to this known directory must be chnged in the module source at the three "open" statements.

Release 1

SAVS

IRI_F(6)

NAME

MSIS_F - Neutral Atmosphere Empirical Model from the surface to lower exosphere

SUMMARY

Name MSIS (F)

Type data input

Inputs field 3-space float

Outputs field 3-space float

Parameters	Name	Type	Default	Min	Max
	run	oneshot	0	0	1
	IYD	integer	1	1	365
	UT	float	12.	0.	24.
	F107A	float	100.	0.	200.
	F107	float	100.	0.	200.
	AP	float	4.	0.	50.

DESCRIPTION

Neutral Atmosphere Empirical Model from the surface to lower exosphere MSISE90 (JGR, 96, 1159-1172, 1991)

AUTHOR OF MSIS STAND ALONE CODE:

A.E.Hedin

F107, F107A, and AP effects are not large below 80 km and these can be set to 150., 150., and 4. respectively.

O, H, and N set to zero below 72.5 km Exospheric temperature set to average for altitudes below 120 km.

INPUTS

Field (field 3-space float; REQUIRED)

The input field defines the seed points for the MSIS model. It is at this field's coordinates that the MSIS will calculate the desired output. The MSIS can accept rectilinear OR irregular point meshes (and output the same) from any source as long as the coordinate information is 3-space in a spherical system (*i.e.*, LAT/LONG/ALT). It is for this reason that the module "Starter Points" is ideal input for the MSIS model module.

PARAMETERS

run - depress to execute MSIS model and send desired output to output port

IYD - day of year

UT - universal time

F107A - 3 month average of F10.7 flux

F107 - daily F10.7 flux for previous day

AP - magnetic index (daily)

OUTPUTS

Data Field - The output field is 10-vector with the following parameters:

D(1) - HE NUMBER DENSITY(CM-3)
D(2) - O NUMBER DENSITY(CM-3)
D(3) - N2 NUMBER DENSITY(CM-3)
D(4) - O2 NUMBER DENSITY(CM-3)
D(5) - AR NUMBER DENSITY(CM-3)
D(6) - TOTAL MASS DENSITY(GM/CM3)
D(7) - H NUMBER DENSITY(CM-3)
D(8) - N NUMBER DENSITY(CM-3)
T(1) - EXOSPHERIC TEMPERATURE
T(2) - TEMPERATURE AT ALT

Release 1

SAVS

MSIS_F(6)

NAME

WINDS_F - horizontal winds covering all altitude regions

SUMMARY

Name WINDS (F)

Type data input

Inputs field 3-space float

Outputs field 3-space float

Parameters	Name	Type	Default	Min	Max
	run	oneshot	0	0	1
	IYR	integer	90	0	99
	IDAY	integer	1	1	365
	UT	float	12.	0.	24.
	F107A	float	100.	0.	200.
	F107	float	100.	0.	200.
	AP	float	4.	0.	50.

DESCRIPTION

AUTHOR OF WINDS STAND ALONE MODEL:

A. E. Hedin

This model represents the horizontal winds covering all altitude regions. The module places a three component vector at each seed point. These components are the meridional, zonal, and zero values of the wind vector.

INPUTS

Field (field 3-space float; REQUIRED)

The input field defines the seed points for the WINDS model. It is at this field's coordinates that the WINDS will calculate the desired output. The WINDS can accept rectilinear OR irregular point meshes (and output the same) from any source as long as the coordinate information is 3-space in a spherical system (*i.e.*, LAT/LONG/ALT). It is for this reason that the module "Starter Points" is ideal input for the WINDS model module.

PARAMETERS

run - depress to execute WINDS model and send desired output to output port

IYR - two digit integer representing year

IDAY - day of year

UT - universal time

F107A - 3 month average of F10.7 flux

F107 - daily F10.7 flux for previous day

AP - magnetic index (daily)

OUTPUTS

Data Field - The output field is 3-vector:

W(1) - meridional (m/sec), where north is positive

W(2) - zonal (m/sec), where east is positive

W(3) - 0 (since the output is horizontal winds)

Release 1

SAVS

WINDS_F(6)

NAME

auroral_oval - Feldstein auroral oval model

SUMMARY

Name auroral_oval

Type data input

Inputs NONE

Outputs boundry1: field 1D 3-space 0-vector irregular float
boundry2: field 1D 3-space 0-vector irregular float

Parameters	Name	Type	Default	Min	Max
	q_value	integer	4	0	8
	UT	float	12.	0.	24.
	doutput	choice	LAT LON ALT LAT LON ALT: X Y Z		

DESCRIPTION



This module generates two output fields. Each field represents the boundaries of the Feldstein oval in a hemisphere. These boundaries are lists of points (either in spherical or rectangular coordinates, depending on the selection of the doutput parameter), to be connected to the "scatter dots" module for visualization.

PARAMETERS

q_value - Q value

UT - universal time (in decimal hours)

doutput - If "LAT LON ALT" is selected, the output points will be spherical coordinates. Otherwise, if "X Y Z" is selected, the output points will be rectangular coordinates.

OUTPUTS

Each output field is a list of points that make up the boundaries of the oval; one output field for each hemisphere. The coordinate system of the output points depends on the value of doutput.

Release 1

SAVS

auroral_oval(6)

NAME

trace_fieldline - traces B-field lines from s/c positions

SUMMARY

Name	trace_fieldline				
Type	mapper				
Inputs	in_fld:	field 1D 3-space irregular float	REQUIRED		
	times_UT:	field 1D 5-vector uniform integer	REQUIRED		
	region_colors:	field 1D 3-vector uniform float	OPTIONAL		
Outputs	fld_pts:	field 1D 3-space 0-vector irregular float			
	field_line_geom:	geom polytri			
Parameters	Name	Type	Default	Min	Max
	data_type	choice	GEO	GEO:GSE:GSM	
	Kp_level	integer	1	1	6
	clear_lines	toggle	1	0	1

DESCRIPTION

C This module uses the Tsyganenko_89 code (TRACE and RECALC) to trace field
 C lines from s/c positions. The outputs are a field listing the coordinates
 C of the field line that passes through the first seed point and a Polytri
 C geometry containing one polyline for each input position that was traced.

C

C Written by:

C

C David McNabb

C Advanced Visualization Lab

C University of Maryland at College Park

C

C 4/2/92

C

C VERSION 1.1 6/15/94 Paul E. Blanchard

C Input point(s) (into "in_fld" port) can be:

C XPOS YPOS ZPOS

C GEO - Geographic coords -90->90 (deg) 0->360 (deg) (m/Km/Re)

C GSE - Solar Ecliptical coords

C GSM - Solar Magnetospheric coords

INPUTS

in_fld (field 1D 3-space irregular float; REQUIRED)

This input field defines the seed points for trace_fieldlines. At each seed point, trace_fieldlines will draw a B-field line that passes through the point and extends to the Earth and/or magnetosphere. "Starter Points" module is ideal for this input (in a LAT/LON/ALT coordinate system).

times_UT (field 1D 5-vector uniform integer; REQUIRED)

This input supplies the time components to module (*i.e.* year, day-in-year, hour, minute, and second). Use module "time_UT" for this input.

region_colors (field 1D 3-vector uniform float; OPTIONAL)

This input simply supplies trace_fieldline with eight colors to color the lines according to magnitude. Use module "eight colors" for this input.

PARAMETERS

data_type - defines the coordinate system of the input points

GEO -> Geographical coords

GSE -> Solar Ecliptical coords

GSM -> Solar Magnetospheric coords

Kp_level - Kp value

clear_lines - when on, reinitializes AVSgeom_edit_list (if need be). For the most part don't worry about this parameter.

OUTPUTS

fld_pts (field 1D 3-space 0-vector irregular float)

This field contains coordinates of the field line that passes through the first seed point.

field_line_geom (geom polytri)

This is the geometry output port that has the object information necessary to draw the field lines. This port gets connected directly to the geometry viewer.

Release 1

SAVS

trace_fieldline(6)

NAME

Field Of View - produces "flashlight" beam from a point source

SUMMARY

Name	Field Of View				
Type	data input				
Inputs	NONE				
Outputs	out: field 2D 3-space 1-vector irregular float off: field 1D 3-space 0-vector irregular float				
Parameters	Name	Type	Default	Min	Max
	plat	float	18.5	-90.	90.
	plon	float	293.	0.	360.
	max_alt	float	500.	0.	10000.
	half_ns	float	0.125	0.	3.14
	half_ew	float	0.0625	0.	3.14

DESCRIPTION

This module originates the "field of view" at the latitude/longitude position given in the parameters (altitude is 0. kilometers). It calculates the "field of view" up to the "max_alt" along a radial line from the center of the Earth.

In the source, the altitude step is hardwired to 10 Km and at each altitude step there are 37 points around the "field of view".

PARAMETERS

plat - latitude position of "field of view" origin (in degrees)

plon - longitude position of "field of view" origin (in degrees)

max_alt - maximum altitude of "field of view" (in Km)

half_ns - half angle of "field of view" in north/south direction (in rad)

half_ew - half angle of "field of view" in east/west direction (in rad)

OUTPUTS

out - list of points that defines the "field of view" structure

off - contains the point source point (in degrees)

APPENDIX - FILTERS

SAVS Modules

set minmax(6)

NAME

set minmax - sets min/max of input field

SUMMARY

Name	set minmax
Type	filter
Inputs	field scalar float field scalar float
Outputs	field scalar float field scalar float float float

Parameters	Name	Type	Default	Min	Max
FIELD INFO	string_block				
calc	choice	internally set	internally set	hardwire :	
min	float	0	unbound	unbound	
max	float	1	unbound	unbound	

DESCRIPTION

The "set minmax" module allows the user to set the min/max field attribute of the input field(s), and pass this to the output field(s). If "internally set" is chosen, the overall min/max from the field(s) is passed to the output field(s). If "hardwire" is selected, the min and max dials appear and allows the user to set the min/max of the output field(s). The data is not altered in any way.

INPUTS

Data Field 1 (required; field scalar float)
This input data must have a single float value at each coordinate.

Data Field 2 (optional; field scalar float)
This input data must have a single float value at each coordinate.

PARAMETERS

FIELD INFO

This panel displays the calculated min/max from the input field(s)

calc

If "internally set" is selected, then the "min" and "max" dials are hidden and the overall min and max displayed in "FIELD INFO" are assigned to the min/max field attribute of the output field(s).

If "hardwire" is selected, then the "min" and "max" dials show and the user-defined values from the dials are assigned to the min/max field attribute of the output field(s).

min
float value defining the minimum data value

max
float value defining the maximum data value

OUTPUTS

Output Field 1
identical to "Data Field 1", except that it has new/revised min/max attribute

Output Field 2
identical to "Data Field 2", except that it has new/revised min/max attribute

float 1
this float value contains "min" value

float 2
this float value contains "max" value

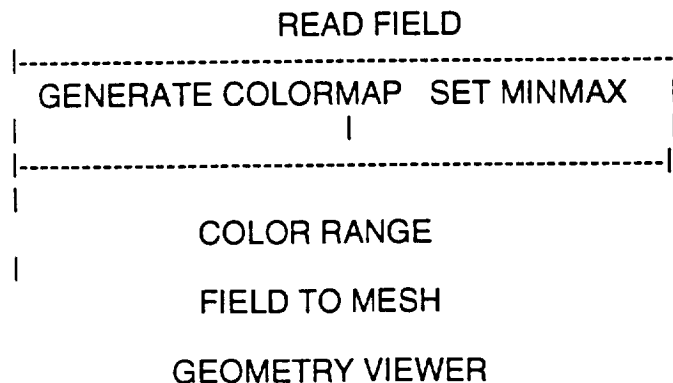
NOTES

This module comes in handy when the user is visualizing a time history of data, and wants to keep the color scheme the same through out the time range. It also works well if the user is comparing two fields and wants the color scheme (range) to be the same for each field.

The two fields do not have to be of the same size or type.

EXAMPLE

Through a time history (varying data ranges), this example shows how the user can keep the same color scheme (by choosing "hardwire").



Release 1

SAVS

set minmax(6)

NAME

downsize - reduce size of data set by sampling

SUMMARY

Name	downsize				
Type	filter				
Inputs	field 1D/2D/3D				
Outputs	field of same type as input				
Parameters	Name	Type	Default	Min	Max
	downsize	integer	8	1	16

DESCRIPTION

The downsize module changes the size of the input data set by subsampling the data. It extracts every Nth element of the field along each dimension, where N is the value of the downsize factor parameter. This technique preserves the aspect ratio of the input data.

This module is useful for operating on a reduced amount of data, in order to adjust other processing parameters interactively, or save memory. After the parameter values have been set, you can remove the downsize module, so that the full data set is used for final processing.

Alternatively, retain the downsize module in the network, so that you can interactively choose between image quality (downsize factor = 1 for highest-resolution data) and execution speed (downsize factor > 1 for lower-resolution data).

INPUTS

Data Field (required; field 1D/2D/3D)
The input data may be any AVS field.

PARAMETERS

downsize
Determines how data elements from the field are sampled. Increasing this parameter causes more elements to be skipped over, thus decreasing the size of the output.

OUTPUTS

Data Field
The output field has the same dimensionality as the input field, but the number of elements in each dimension is reduced by the downsize factor.

The min_val and max_val attributes of the output field are invalidated. Note that the extent is unmodified; this module changes the resolution of the data within the

physcial space delimited by the extents. It does not alter the physical extents of the data.

Release 1

SAVS

downsize(6)

NAME

crop - extract subset of elements from a field

SUMMARY

Name crop

Type filter

Inputs field 1D/2D/3D

Outputs field of same type as input

Parameters	Name	Type	Default	Min	Max
	min x	int	1st indx	1st indx	ast indx
	max x	int	last indx	1st indx	last indx
	min y	int	1st indx	1st indx	last indx
	max y	int	last indx	1st indx	last indx
	min z	int	1st indx	1st indx	last indx
	max z	int	last indx	1st indx	last indx
	size to fit	toggle	off	off	on

DESCRIPTION

The crop module changes the size of a field by extracting the data within a specified range of elements. This process is analogous to "cropping" a photographic image.

This module is useful for subsampling the data without changing it (*e.g.* by interpolation). It preserves the resolution of the data, but may change its aspect ratio. Typical uses are to eliminate uninteresting portions of the data and to increase processing speed by reducing the amount of data.

Once a field is input to crop, the module's min and max dials are set to the min and max indices of the field's data array. From then on the dials cannot be turned lower than the min index, or higher than the max index (min cannot equal max, and min must be less than or equal to max-1). If you use the Dial Editor to change these values they would "snap back" to their original values. This makes sense, because you can only take a subset from the field within the field's array indices.

When the minx and max dials are set to the same value, crop first tries to set max=max+1. If max is already at its maximum index, then crop sets min=min-1.

INPUTS

Data Field (required; field 1D/2D/3D)

The input data may be any AVS field.

PARAMETERS

Note that the parameters indicate positions of elements in the field - they have nothing to do with the values of field elements.

min x

Specifies the lower bound array index in the field's first dimension.

max x

Specifies the upper bound array index in the field's first dimension.

min y

(Does not appear for 1D input data sets) Specifies the lower bound array index in the field's second dimension.

max y

(Does not appear for 1D input data sets) Specifies the upper bound array index in the field's second dimension.

min z

(Does not appear for 1D or 2D input data sets) Specifies the lower bound array index in the field's third dimension.

max z

(Does not appear for 1D or 2D input data sets) Specifies the upper bound array index in the field's third dimension.

size to fit

(Does not appear for 1D input data sets) In the default mode, crop does not change the extent information in the output field structure. This is because you may wish to merge cropped interpretations of a data with interpretations of the original data. By retaining the original extent information, the cropped version of the data causes the Geometry Viewer to adjust its extents appropriately. The points array contains the actual cropping information. With uniform fields, when the size to fit button is turned on, the points array gets copied to the extents array which has the effect of causing the Geometry Viewer to scale the window to exactly fit the cropped data rather than the extents of the original, uncropped data.

OUTPUTS

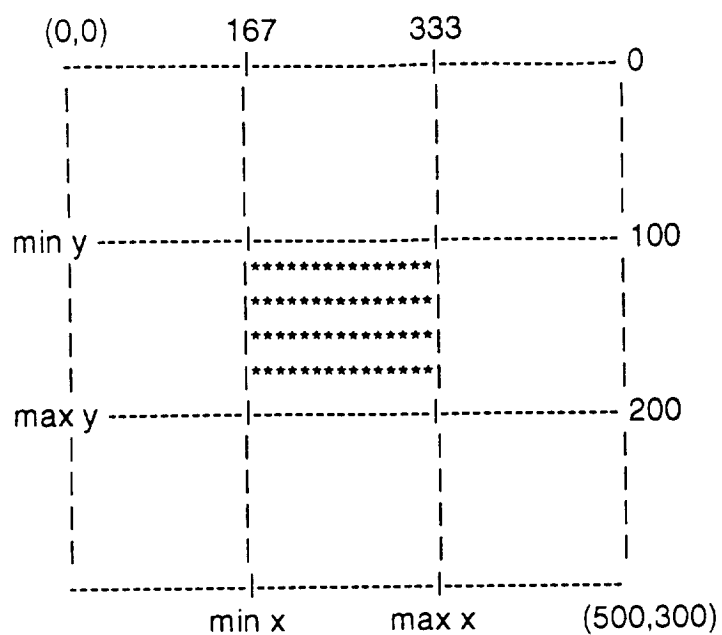
Data Field

The output field has the same dimensionality as the input field, but the number of elements in each dimension is reduced.

The min_val and max_val attributes of the output field are invalidated.

EXAMPLE

Suppose you want to process the middle third of a field that contains an 500x300 pixel image:



Set the x-axis and y-axis parameters as follows:

min x = 167
max x = 333
min y = 100
max y = 200

Release 1

SAVS

crop(6)

NAME

LLA_to_XYZ - spherical to rectangular coordinate transformation

SUMMARY

Name LLA_to_XYZ

Type filter

Inputs field 3-space float

Outputs field 3-space irregular float

Parameters	Name	Type	Default	Min	Max
	----	----	-----	---	---
	delta	typein real	100	0	200
	alt_unit	choice	kilometers	meters:kilometers:earth	

radii

DESCRIPTION

For this module to work, the input field coordinate points must be in the ordered triple : (latitude, longitude, altitude). The altitude point can be in meters, kilometers, or earth radii (the choice is set by the "alt_unit" parameter). The module converts this ordered triple from spherical coordinates to rectangular coordinates.

INPUTS

Data Field (required; field 3-space float)

This input data can have multiple data values at each coordinate, but the coordinates must be mapped into 3-D space. This module can handle rectilinear or irregular.

PARAMETERS

delta

This real value is a constant multiplied to each transformation coordinate, based on 1 Re (*i.e.*, 100% multiplies each coordinate by 1, 150% multiplies each coordinate by 1.5).

alt_unit

This choice identifies the input altitude units

OUTPUTS

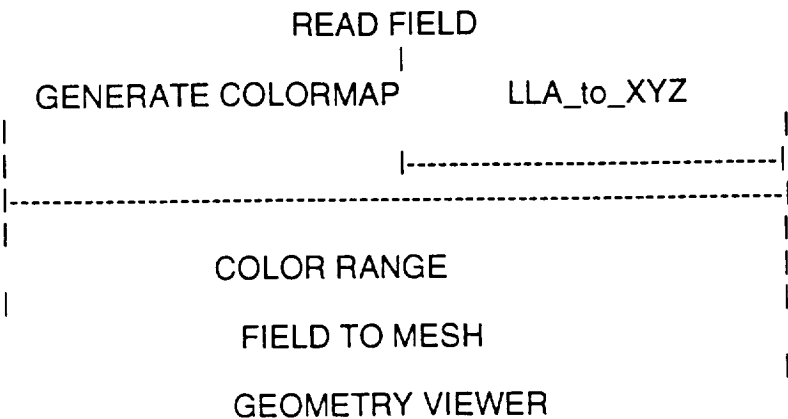
Output Field 1

Output data is identical to input data. The output coordinate information is always irregular and mapped into rectangular space.

NOTES

This module is used to visualize latitude/longitude/altitude data on a globe. It is up to the user to indetify the units of the altitude inputs.

EXAMPLE



RELATED MODULES

Vspher2Vrect

Release 1

SAVS

LLA_to_XYZ(6)

NAME

Vspher2Vrect - spherical to rectangular coordinate transformation

SUMMARY

Name Vspher2Vrect

Type filter

Inputs field 3-space 3-vector float

Outputs field 1D 3-space 3-vector irregular float

Parameters	Name	Type	Default	Min	Max
	----	----	-----	---	---
	alt_unit	choice	kilometers	meters:kilometers:earth	

radii

DESCRIPTION

For this module to work, the input field coordinate points must be in the ordered triple : (latitude, longitude, altitude). The altitude point can be in meters, kilometers, or earth radii (the choice is set by the "alt_unit" parameter). The module converts this ordered triple from spherical coordinates to rectangular coordinates.

The 3 components of the vector supplied by the input field are:
 north/south component (northward is positive)
 east/west component (eastern is positive)
 radial component (outward from center of earth)
 The units of the vector don't matter.

INPUTS

Data Field (required; field 3-space 3-vector float)

This input data must be a 3-vector at each coordinate. The first component of the input vector is the north/south, the second is east/west, and the third is the radial. The input coordinate is the node at which the vector is based (in a latitude/longitude/ altitude coordinate frame). This module can handle rectilinear or irregular input fields.

PARAMETERS

alt_unit

This choice identifies the input altitude units

OUTPUTS

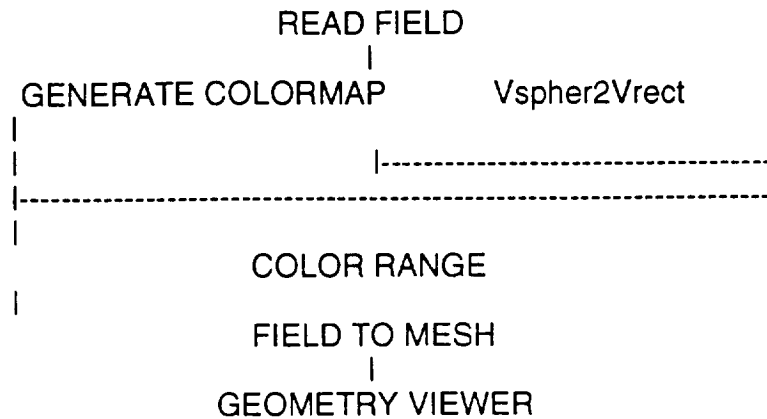
Output Field 1

Output data is proportional to input data. The output coordinate information is always irregular and mapped into rectangular space.

NOTES

This module is used to visualize latitude/longitude/altitude data on a globe. It is up to the user to identify the units of the altitude inputs.

EXAMPLE



RELATED MODULES

LLA_to_XYZ

Release 1

SAVS

Vspher2Vrect(6)



NAME

fill_points - linearly interpolates a user-definable number of points between fixed input points

SUMMARY

Name	fill_points				
Type	filter				
Inputs	field 1D 3-space irregular float				
Outputs	field 1D 3-space irregular float				
Parameters	Name	Type	Default	Min	Max
	fill_pts	integer	0	0	50

DESCRIPTION

This module increases the size of the input data set. It adds N points between every two input points by linearly interpolation, where N is the value of the fill_pts parameter. This technique preserves the aspect ratio of the input data.

INPUTS

Data Field (required; field 1D 3-space irregular float)
The input data may be thought of as a line through space.

PARAMETERS

fill_pts
Determines how many points to linearly interpolate (add) between input points. Increasing this parameter causes more elements to be written to output field. If the number of input points is NUM, the number of output points is: NUM + fill_pts*(NUM-1).

OUTPUTS

Data Field
If the number of input points is NUM, the number of output points is: NUM + fill_pts*(NUM-1).

NAME
line_integral - linear interpolation of input field

SUMMARY						
Name	linr_integral					
Type	filter					
Inputs	field 3-space irregular float					
Outputs	field 3-space irregular float					
Parameters	Name	Type	Default	Min	Max	
NONE						

DESCRIPTION

This module integrates the input data set by adding (distance between points I and J)*(average data value of I and J) to data value I. This is a fast-no-frills interpolation.

INPUTS

Data Field (required; field 3-space irregular float)
The input data may be thought of as a line through space.

PARAMETERS

NONE

OUTPUTS

Data Field
The value at each coordinate represents the integrated sum of the input field up to that point.

Release 1	SAVS	line_integral(6)
-----------	------	------------------

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE October 31, 1994		3. REPORT TYPE AND DATES COVERED Contractor Report
4. TITLE AND SUBTITLE A Space and Atmospheric Visualization Science System			5. FUNDING NUMBERS 930	
6. AUTHOR(S) E.P. Szuszczewicz, P. Blanchard, and A. Mankofsky				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Science Applications International Corporation 1710 Goodridge Drive McLean, VA 22102			8. PERFORMING ORGANIZATION REPORT NUMBER NAS5-32337 5555-12	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration - HQ/ OSSA Washington, D.C. 20546-0001 Universities Space Research Association 10227 Wincopin Circle, Suite 212 Columbia, MD 21044			10. SPONSORING/MONITORING AGENCY REPORT NUMBER CR-189390	
11. SUPPLEMENTARY NOTES Technical Monitor: J. Hollis, Code 930				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified-Unlimited Subject Category 82 Report is available from the NASA Center for AeroSpace Information, 800 Elkridge Landing Road, Linthicum Heights, MD 21090; (301) 621-0390.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) SAVS (a Space and Atmospheric Visualization Science System) is an integrated system with user-friendly functionality that employs a "push-button" software environment that mimics the logical scientific processes in data acquisition, reduction, analysis, and visualization. All of this is accomplished without requiring a detailed understanding of the methods, networks, and modules that link the tools and effectively execute the functions. This report describes SAVS and its components, followed by several applications based on generic research interests in interplanetary and magnetospheric physics (IMP/ISTP), active experiments in space (CRRES), and mission planning focused on the Earth's thermospheric, ionospheric and mesospheric domains (TIMED). The final chapters provide a user-oriented description of interface functionalities hands-on operations, and customized modules, with details of the primary modules presented in the Appendices. The overall intent of the report is to reflect the accomplishments of the three-year development effort and to introduce potential users to the power and utility of the integrated data acquisition, analysis and visualization system.				
14. SUBJECT TERMS Visualization, Data Acquisition, Space Science			15. NUMBER OF PAGES 80	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT Unlimited	